LEONARDO AUGUSTO MARTUCCI

DOMÍNIOS VIRTUAIS PARA REDES MÓVEIS AD HOC:

UM MECANISMO DE SEGURANÇA

Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do Título de Mestre em Engenharia

São Paulo

2002

LEONARDO AUGUSTO MARTUCCI

DOMÍNIOS VIRTUAIS PARA REDES MÓVEIS AD HOC:

UM MECANISMO DE SEGURANÇA

Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do Título de Mestre em Engenharia

Área de Concentração:
Sistemas Digitais

Orientador:
Profa. Dra. Tereza Cristina Melo de Brito Carvalho

São Paulo

2002

"'Would you tell me, please, which way I ought to go from here?'

'That depends a good deal on where you want to get to,' said the Cat.

'I don't much care where-' said Alice.

'Then it doesn't matter which way you go,' said the Cat."

Alice's Adventures in Wonderland
Lewis Carrol

Ao meu avô e a minha avó,
Moacyr e Dirce Martucci,
pelo amor ao seu filho, Moacyr

À memória de
Masashi e Takako Sakatsume
pelo amor à sua filha, Olga

AGRADECIMENTOS

Este trabalho, apesar de escrito por um só homem, é o resultado da vontade e esforços de diversas pessoas, à quais devo minha eterna gratidão:

Aos meus pais, Moacyr e Olga Martucci, pelo amor e apoio incondicionais.

À Dra. Candice Lima e ao Dr. Agnaldo Anelli, minha mais profunda gratidão, pela sabedoria, ternura e dedicação exemplar a uma missão reservada a poucos: conceder esperança aos que necessitam dela. Meus agradecimentos também à inesquecível equipe do Hospital do Câncer A.C.Camargo.

Aos doutores Milton André Dantas, Luís Cardoso e Noedir Stolf, assim como à equipe médica do Instituto do Coração, pelo seu fabuloso trabalho.

Aos amigos, e também companheiros no planejamento e execução do projeto que deu origem a este trabalho, especialmente: Yeda Regina Venturini, pelas intermináveis discussões sobre soluções e destinos a serem tomados pelo projeto; Armin Werner Mittelsdorf, pelo levantamento do problema que deu origem ao mecanismo proposto neste trabalho; Fernando Frota Redigolo, pela sua experiência e colaboração na análise estatística; e Christiane Marie Schweitzer, pelo seu equilíbrio e inestimável apoio à realização deste trabalho.

À amiga Luciana Zaina e aos amigos Pedro Mindlin e Oscar Vilcachagua, pela colaboração nos aspectos relacionados ao projeto orientado a objetos e ao ambiente de programação. À Ana Coracini, inabalável em sua força.

À Profa. Dra. Tereza Cristina Melo de Brito Carvalho, pelo seu incentivo, apoio e orientação, determinantes para a conclusão desta dissertação e ao Prof. Dr. Wilson Vicente Ruggiero, pelo seu apoio e impecável trabalho de coordenação.

Agradeço a dedicação de Lindaura Costa e a amizade de meu irmão, Daniel F. Martucci, assim como de meus demais irmãos, que conheci ao longo de minha vida: Paulo de Andréa, Carlos E. Santoro (Bidu), G. Marcel Smetana, Bruno Galiotto e Fernando Szterling.

RESUMO

As redes de computadores móveis ad hoc, ou seja, redes formadas por dispositivos móveis que se movimentam de modo arbitrário, estabelecendo enlaces de comunicação automaticamente, constituem uma rede efêmera sem a presença de entidades centrais. Estas características peculiares impedem a utilização das técnicas e dos protocolos empregados nas redes estruturadas, de modo que se torna necessário o desenvolvimento de mecanismos adequados para as redes móveis ad hoc.

Esta dissertação, denominada “Domínios Virtuais para Redes Móveis Ad Hoc: um Mecanismo de Segurança” apresenta a proposta e a implementação do protótipo de um mecanismo que possa assegurar alguns aspectos da privacidade em uma rede desta natureza, de modo a garantir que informações relativas à composição ou aos serviços existentes nesta, assim como informações relativas à sua movimentação, não possam ser obtidas por uma parte não-autorizada.

O mecanismo procura garantir a segurança dos dispositivos organizando-os em domínios virtuais, que correspondem a grupos organizados em torno de uma característica comum, ou seja, pertencem a um mesmo usuário ou ainda são utilizados em um mesmo ambiente, sendo seus participantes considerados confiáveis. Cada domínio virtual é identificado por uma seqüência de valores produzida por um gerador de números pseudo-randômicos, atualizada todo milésimo de segundo, que utiliza como semente uma informação compartilhada, comum aos dispositivos pertencentes a um domínio virtual, e um índice de tempo.

Esta dissertação apresenta inicialmente os requisitos funcionais a serem atendidos por um mecanismo de segurança adequado às redes móveis ad hoc e trabalhos relacionados ao mecanismo proposto. A arquitetura do mecanismo é então descrita, seguida de uma especificação detalhada dos blocos que a compõem. A implementação do protótipo, assim como a realização de testes sobre cada um blocos de seus blocos são apresentados ao longo desta dissertação.

ABSTRACT

Wireless ad hoc computer networks are a group of wireless nodes that can move arbitrarily, establishing communication links automatically and, therefore, building an ephemera network without the presence of any central entity. The usage of known techniques and protocols applied in regular networks is not possible in wireless ad hoc networks due to its very particular characteristics, and the development of new mechanisms suitable to these networks is required.

This master thesis, called “Virtual Domains for Wireless Ad Hoc Networks: a Security Mechanism” presents the proposal and the prototype implementation of a mechanism that can assure some privacy aspects on wireless ad hoc networks, guarantying that information regarding to the network composition, the existing services or its location cannot be obtained from non-authorized parties.

The devices are organized in virtual domains, which are groups of devices that share a common characteristic, as belonging to the same owner or to the same environment. Moreover, all members of a virtual domain are considered trustable devices. Each virtual domain is defined by an output sequence. This sequence is produced by a pseudo-random number generator, updated every millisecond. The seed of the pseudo-random number generator is composed by shared information and time value.

This master thesis initially presents the functional requirements related to a security mechanism suitable for wireless ad hoc networks and its related research works. Furthermore, the security mechanism architecture is presented as well as a detailed specification of its building blocks. Finally, the prototype implementation and the tests performed to evaluate its functionalities are presented.

SUMÁRIO

Capítulo 1 Introdução 1

1.1 Objetivos 2

1.2 Motivação 2

1.3 Escopo 4

1.4 Organização do Trabalho 5

Capítulo 2 Requisitos e Trabalhos Relacionados 8

2.1 Aspectos de Segurança em Redes Móveis Ad Hoc 9

2.2 Questões de Segurança em Redes Ad Hoc 10

2.3 Procura pela Segurança em Redes Ad Hoc 11

2.3.1 Modelo de Segurança Resurrecting Duckling 11

2.3.2 Entidade Certificadora Distribuída de Zhou e Haas 13

2.3.3 Projeto Terminodes e Certificados em Cadeia 15

2.3.4 Anonimato como Aspecto da Privacidade 17

2.3.5 Questão da Geração e Distribuição de Chaves 18

2.3.6 Outras Propostas para a Questão da Autenticidade 19

2.3.6.1 Direcionada a um protocolo de roteamento 19

2.3.6.2 Fundamentado na confiança 20

2.3.6.3 Estendendo o modelo Resurrecting Duckling 21

2.3.6.4 Utilizando chaves públicas 21

2.3.7 Modelo para Redes Orientadas a Serviços 22

2.3.8 Considerações 23

Capítulo 3 O Mecanismo - Domínios Virtuais 24

3.1 Requisitos Funcionais 25
3.2 Domínios de Redes Virtuais 25
3.2.1 Endereçamento dos Domínios Virtuais 26
3.2.2 Funcionamento do Mecanismo 26
3.2.2.1 A inspiração 27
3.2.2.2 Um mecanismo semelhante – RSA SecurID 28
3.2.2.3 Seqüência de troca das mensagens 28
3.2.2.4 Posicionamento do mecanismo 30
3.2.2.5 Formato das mensagens 31
3.2.2.6 Fluxogramas 33
3.2.2.7 Considerações sobre a troca das mensagens 36
3.3 Arquitetura do Mecanismo de Domínios Virtuais 36
3.3.1 Gerador de Números Pseudo-aleatórios 38
3.3.1.1 PRNG ANSI X9.31 38
3.3.1.2 PRNG DSA 40
3.3.1.3 PRNG RSAREF 41
3.3.1.4 PRNG Blum Blum Shub 42
3.3.1.5 PRNG Yarrow-160 43
3.3.1.6 Considerações sobre os PRNG 44
3.3.2 As Sementes 45
3.3.3 As Relações Temporais e o Relógio do Sistema 46
3.3.4 Verificação da Origem 48
3.3.5 O Gerente e a Tabela de Mensagens Ativas 49
3.3.6 Transmissor/Receptor 50
3.4 O Mecanismo em seu Contexto 51

3.4.1 Chaves Públicas: Autenticação e Anonimato 51
3.4.1.1 A criação do anonimato completo 51
3.4.1.2 Considerações sobre o consumo de energia 52
3.4.2 A Geração e Distribuição das Sementes 53
3.4.3 Renovação Automática das Sementes 53
Capítulo 4 Implementação 55
4.1 Aspectos Gerais da Implementação 55
4.1.1 O Escopo da Implementação 55
4.1.2 Ambiente e Linguagem de Programação 56
4.1.3 Interface do Mecanismo 57
4.1.4 Redes Wireless 58
4.2 Máquina de estados 58
4.3 Campos das Mensagens 60
4.3.1 Tipos de Mensagens 60
4.3.2 Número de Seqüência 61
4.3.3 Tempos 62
4.3.4 Valores Gerados 62
4.4 Casos de Uso 63
4.5 Especificação de Classes 64
4.5.1 Diagrama de Classes 65
4.5.2 Descrição das Classes 65
4.5.2.1 Classe AuthNetwork 65
4.5.2.2 Classe SeedTable 66
4.5.2.3 Classe MessageIndex 67
4.5.2.4 Classe PRNG 68

4.6 Diagramas de Interação 68
4.6.1 Alteração de Parâmetros do Mecanismo 68
4.6.1.1 Alteração da porta TCP 68
4.6.1.2 Redefinição do tamanho da janela de tempo 69
4.6.1.3 Inclusão de um domínio virtual 69
4.6.1.4 Exclusão de um domínio virtual 70
4.6.1.5 Inclusão de um domínio virtual sem definição explícita da semente 71
4.6.2 Identificação de um Dispositivo 72
4.7 Considerações sobre a Implementação 76
Capítulo 5 Testes e Resultados 78
5.1 Metodologia de Testes Utilizada 78
5.2 O PRNG do Mecanismo de Domínios Virtuais 79
5.2.1 Produção de Amostras 80
5.2.2 Teste χ^2 81
5.2.2.1 Testes das amostras de 10^4 valores 82
5.2.2.2 Testes das amostras de 100k valores 88
5.2.2.3 Conclusões sobre os resultados dos testes χ^2 92
5.2.3 Outros Testes 93
5.2.3.1 Teste de freqüência (monobit) 94
5.2.3.2 Teste serial (two-bit test) 95
5.2.3.3 Teste poker (poker test) 96
5.2.3.4 Teste de comprimento de seqüências (runs test) 97
5.2.3.5 Conclusões sobre os testes realizados 99
5.2.4 Bateria de Testes FIPS PUB 140-2 99

5.2.4.1 Monobit 100
5.2.4.2 Poker test 100
5.2.4.3 Runs test 101
5.2.4.4 Long run test 102
5.2.4.5 Conclusões obtidas da bateria de testes FIPS 140-2 .. 103
5.2.5 Conclusões sobre os testes realizados sobre o PRNG..... 103
5.3 Operações sobre Informações Armazenadas 103
5.3.1 Inclusão e Exclusão de Sementes 104
5.3.1.1 Inclusão de sementes. 104
5.3.1.2 Exclusão de uma semente. 105
5.3.2 Inserção e Remoção de Mensagens 106
5.3.2.1 Inclusão de uma mensagem 107
5.3.2.2 Exclusão de uma mensagem 108
5.4 Mecanismo de Domínios Virtuais 109
5.4.1 Configurando Domínios Virtuais 110
5.4.1.1 Inclusão automática de um domínio virtual..... 111
5.4.1.2 Inclusão de um domínio virtual 111
5.4.1.3 Exclusão de um domínio virtual 112
5.4.2 Alteração de Parâmetros 113
5.4.2.1 Porta TCP 113
5.4.2.2 Janela de tempo..... 114
5.4.3 Reconhecimento de um Domínio Virtual..... 114
5.4.3.1 As janelas de tempo 118
5.4.3.2 Não reconhecimento de um domínio virtual..... 119
5.5 Considerações sobre os Testes e Resultados 120

Capítulo 6 Considerações Finais 122

6.1 Discussão sobre os Resultados Obtidos 122

6.2 Contribuições e Posicionamento do Trabalho 123

6.3 Continuidade 125

Lista de Referências 126

Bibliografia Recomendada 135

LISTA DE FIGURAS

Figura 3.1: Reconhecimento de um domínio virtual. 28
Figura 3.2: Posicionamento do mecanismo de Domínios Virtuais. 30
Figura 3.3: Formato de mensagem padrão utilizada pelo mecanismo. 31
Figura 3.4: Mensagem *desafio*. 32
Figura 3.5: Mensagem *resposta*. 32
Figura 3.6: Mensagem *réplica*. 33
Figura 3.7: Fluxograma da procura por um domínio virtual. 34
Figura 3.8: Fluxograma da requisição de um domínio virtual. 35
Figura 3.9: Arquitetura do mecanismo de formação de Domínios Virtuais. 36
Figura 3.10: Janela de tempo do dispositivo A. 46
Figura 3.11: Intervalo mínimo de uma janela de tempo. 47
Figura 3.12: Tempo de validade mínimo para uma mensagem enviada. 50
Figura 4.1: Máquina de estados do mecanismo de Domínios Virtuais. 59
Figura 4.2: Troca de mensagens e os campos *tipo* e *seqüência*. 61
Figura 4.3: Diagrama de classes do mecanismo de Domínios Virtuais. 65
Figura 4.4: Alteração da porta TCP. 69
Figura 4.5: Redefinição da janela de tempo. 69
Figura 4.6: Inclusão de um domínio virtual. 70
Figura 4.7: Exclusão de um domínio virtual. 71
Figura 4.8: Inclusão de um domínio virtual sem definição da semente. 72
Figura 4.9: Diagrama de seqüência simplificado do processo de identificação de um dispositivo. 75
Figura 4.10: Posicionamento do protótipo desenvolvido. 76
Figura 5.1: Distribuição da amostra *Escritório*. 83

Figura 5.2: Histograma obtido da amostra *Escritório*. 83
Figura 5.3: Distribuição da amostra *Lar doce Lar*. 84
Figura 5.4: Histograma obtido da amostra *Lar doce Lar*. 85
Figura 5.5: Distribuição da amostra *Litoral*. 86
Figura 5.6: Histograma obtido da amostra *Litoral*. 86
Figura 5.7: Distribuição da amostra *Casa de Campo*. 87
Figura 5.8: Histograma obtido da amostra *Casa de Campo*. 88
Figura 5.9: Histograma obtido da amostra *Donovan*. 89
Figura 5.10: Histograma obtido da amostra *Journey*. 90
Figura 5.11: Histograma obtido da amostra *Supertramp*. 91
Figura 5.12: Histograma obtido da amostra *Grassroots*. 92
Figura 5.13: Menu da interface de um objeto da classe *SeedTable*. 104
Figura 5.14: Inclusão de uma semente na tabela. 105
Figura 5.15: Sementes presentes na tabela. 105
Figura 5.16: Exclusão de uma semente da tabela. 105
Figura 5.17: Sementes presentes na tabela após o processo de exclusão. 106
Figura 5.18: Conteúdo da tabela após a exclusão de todas as sementes. . 106
Figura 5.19: Menu da interface de um objeto da classe *MessageIndex*. ... 106
Figura 5.20: Operações sobre mensagens da Tabela Desafio. 107
Figura 5.21: Inclusão de uma mensagem na tabela. 107
Figura 5.22: Mensagens presentes na tabela após o processo de inclusão. 108
Figura 5.23: Exclusão de uma mensagem da tabela. 108
Figura 5.24: Mensagens presentes na tabela após a operação de exclusão. 108
Figura 5.25: Conteúdo da tabela após a exclusão de todas as mensagens. 109

Figura 5.26: Menu da interface do mecanismo de Domínios Virtuais. 109
Figura 5.27: Opções de configuração dos domínios virtuais. 110
Figura 5.28: Inclusão automática de um domínio virtual. 111
Figura 5.29: Inclusão de um domínio virtual. 111
Figura 5.30: Domínios virtuais cadastrados. 112
Figura 5.31: Remoção de um domínio virtual. 112
Figura 5.32: Domínios virtuais cadastrados após a remoção. 112
Figura 5.33: Menu principal da opção de alteração de parâmetros. 113
Figura 5.34: Redefinição da porta TCP. 113
Figura 5.35: Redefinição da janela de tempo. 114
Figura 5.36: Dispositivos e seus domínios virtuais. 114
Figura 5.37: Domínios virtuais cadastrados no primeiro dispositivo. 115
Figura 5.38: Domínios virtuais cadastrados no segundo dispositivo. 115
Figura 5.39: Dispositivos passam a esperar mensagens. 116
Figura 5.40: Procura por dispositivo pertencente ao domínio *Trabalho*. ... 116
Figura 5.41: Reconhecimento do dispositivo requisitante. 117
Figura 5.42: Reconhecimento de um dispositivo. 118
Figura 5.43: Mensagem recebida fora dos limites da janela de tempo. 119
Figura 5.44: Procurando pelo domínio *Praia*. 119
Figura 5.45: Domínio virtual não reconhecido. 120

LISTA DE TABELAS

Tabela 3.1: PRNG e sua adequação ao mecanismo de Domínios Virtuais...44

Tabela 4.1: Os estados da máquina. 58

Tabela 4.2: Tipos de mensagens e seus códigos. 61

Tabela 4.3: Tipos de mensagens e bits transmitidos de cada valor gerado. 63

Tabela 5.1: Amostras produzidas. 81

Tabela 5.2: Resultados obtidos. 93

Tabela 5.3: Nome das amostras de 2.10^4bits....93

Tabela 5.4: Resultados do teste de freqüência....94

Tabela 5.5: Resultados do teste serial. 96

Tabela 5.6: Resultados dos teste poker. 97

Tabela 5.7: Resultados do teste de comprimento de seqüência. 99

Tabela 5.8: Resultados do teste *monobit* FIPS 140-2. 100

Tabela 5.9: Resultados do *poker test* FIPS 140-2. 101

Tabela 5.10: Limites das taxas de incidência de blocos e lacunas....101

Tabela 5.11: Incidência de blocos e colunas nas amostras. 102

Tabela 5.12: Tamanho dos maiores blocos e lacunas. 102

LISTA DE ABREVIATURAS

3DES	Triple DES
ANSI	American National Standards Institute
AODV	Ad Hoc On-Demand Distance Vector
BD_ADDR	Bluetooth Device Address
CBRP	Cluster Based Routing Protocol
CRL	Certificate Revocation List
DES	Data Encryption Standard
DoS	Denial of Service
DSA	Digital Signature Algorithm
DSR	Dynamic Source Routing
DSS	Digital Signature Standard
FIPS PUB	Federal Information Processing Standards Publication
IARP	Interzone Routing Protocol
IBSS	Independent Basic Service Set
IEEE	Institute of Electrical and Electronics Engineers, Inc
IERP	Intrazone Routing Protocol
IETF	Internet Engineering Task Force
JVM	Java Virtual Machine
LAP	Lower Address Part
MANET WG	Mobile Ad Hoc Networking Working Group
NAP	Non-significant Address Part
NIST	National Institute of Standards and Technology
NTP	Network Time Protocol

PGP	Pretty Good Privacy
PRNG	Pseudo-Random Number Generator
SHA	Secure Hash Algorithm
SL	Serviço de Localização
SR	Serviço de Registro
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TORA	Temporally-Ordered Routing Algorithm
UAP	Upper Address Part
UDP	User Datagram Protocol
UPnP	Universal Plug and Play
UTC	Coordinated Universal Time
ZRP	Zone Routing Protocol

Capítulo 1

Introdução

"The combination of space, time, and strength that must be considered as the basic elements of this theory of defense makes this a fairly complicated matter. Consequently it is not so easy to find a fixed point of departure."

Carl von Clausewitz

As redes ad hoc são formadas por dispositivos móveis que podem mover-se de modo arbitrário, estabelecer enlaces de comunicação automaticamente e, deste modo, criar uma infra-estrutura temporária de comunicação constituindo uma rede efêmera sem a presença de uma entidade central [1].

Segurança é um aspecto de difícil implementação em redes ad hoc devido às particulares características que permitem a sua própria existência, como a natureza esporádica dos enlaces de comunicação, as mudanças freqüentes de topologia e a ausência de entidades centrais, como autoridades certificadoras ou de gerenciamento [2].

Esta dissertação apresenta um mecanismo de segurança a ser aplicado em redes móveis ad hoc, denominado "Domínios Virtuais para Redes Móveis Ad Hoc: um Mecanismo de Segurança", utilizando, para tal, uma identificação de rede que seja variante no domínio do tempo, e capaz de ser integrado com outras soluções e mecanismos de segurança propostos para tais redes.

O restante deste capítulo apresenta, de modo sucinto, os objetivos desta dissertação, a motivação para a sua realização, e o escopo, definindo assim o foco e a abrangência da pesquisa realizada. Finalmente, a estrutura da dissertação é apresentada.

1.1 Objetivos

O objetivo principal desta dissertação é a proposta de um mecanismo de segurança que garanta a privacidade em uma rede móvel ad hoc, de modo a assegurar que:

- Informações relativas à rede, como os dispositivos que a compõem ou os serviços nela disponíveis, não possam ser obtidas por uma parte não-autorizada.
- A movimentação de dispositivos e usuários que a compõem a rede móvel ad hoc não possa ser rastreada por uma parte não-autorizada.

O segundo objetivo a ser alcançado nesta dissertação é a implementação de um protótipo do mecanismo proposto, de modo que se possa verificar a viabilidade de seu desenvolvimento.

Esta dissertação posiciona o mecanismo proposto dentre os demais trabalhos publicados relativos à segurança em redes móveis ad hoc, de modo que a aplicação conjunta destes trabalhos com o mecanismo proposto possa ser capaz de assegurar os diversos aspectos que compõem a segurança para esta particular classe de redes de computadores, respeitando os seus aspectos e requisitos funcionais, sucintamente descritos no item 2.2.

Dentro do universo dos trabalhos publicados analisados nesta dissertação, o mecanismo proposto tem a incumbência de assumir o papel de uma barreira inicial a ataques ativos e criar ainda condições de impedir ataques passivos, através da ocultação da rede móvel.

1.2 Motivação

A utilização de redes móveis ad hoc começou a fazer parte da realidade com o advento das novas tecnologias de redes sem fio e da massificação de dispositivos portáteis com maior capacidade de processamento e interfaces amigáveis.

As redes móveis ad hoc podem ser empregadas em inúmeros ambientes e para diversos fins, sendo, dentre estes, o de maior impacto e com maior potencial de expansão, o de permitir a criação de aplicações ubíquas[1]. Este conceito foi idealizado no final da década de 80 por pesquisadores do XEROX PARC (Palo Alto Research Center) com o intuito de esconder a computação do usuário, de modo a permitir que o usuário não se preocupe com a interface homem-máquina, permitindo que o mesmo se concentre nas interfaces homem-homem [3].

Aplicações ubíquas são aplicações nas quais os usuários são intrinsecamente móveis, a computação está espalhada em todo o ambiente e é invisível para o usuário, através de interfaces altamente amigáveis e intuitivas, permitindo a obtenção de toda e qualquer informação desejada a todo e qualquer momento [4] [5].

A padronização das tecnologias de redes sem fio, como o IEEE 802.11 [6], e a sua conseqüente popularização, em especial o IEEE 802.11b [7], e o surgimento de outras tecnologias com grande potencial de expansão, como o Bluetooth [8], têm alimentado a visão da criação das redes ubíquas, que correspondem a verdadeiros meio-ambientes computacionais, e estimulado a pesquisa e o desenvolvimento de soluções para estas redes nas mais diversas áreas: segurança, roteamento, protocolos de camadas superiores e qualidade de serviço, entre outras.

Como a possibilidade de desenvolvimento de aplicações ubíquas é extremamente vasta, percorrendo desde tradicionais aplicações militares a pequenos itens de uso doméstico e, portanto, de alto impacto social e econômico, padrões passaram então a ser estudados de forma a regulamentar o funcionamento da infra-estrutura ad hoc [4] [5] [9] [10].

Um grupo de trabalho dentro do IETF[2] foi criado, o MANET WG[3], com o objetivo de padronizar um protocolo de roteamento para redes ad hoc [11].

---

[1] **ubíquo**. [Do adv. lat. u*bique*, 'em toda parte', com desin. de adj.] *Adj.*Que está ao mesmo tempo em toda parte; onipresente: *Deus é ubíquo*.
FERREIRA, Aurélio Buarque de Holanda. Novo Dicionário da Língua Portuguesa. 2ª edição. Rio de Janeiro.

[2] *Internet Engineering Task Force*

O único resultado apresentado até o presente momento por este grupo foi a elaboração da RFC2501 [12], de conteúdo apenas informativo.

A pesquisa em redes móveis ad hoc tem abordado, quase que exclusivamente, particularidades e considerações sobre os diversos protocolos de roteamento propostos que tentam se adequar a esta particular categoria de redes de computadores. A maioria dos esforços e recursos tem se destinado a este particular campo da pesquisa em detrimento a outros, como segurança, não menos importantes e que têm recebido, por enquanto, menor atenção da comunidade acadêmica [2].

No entanto, esse panorama tem começado a se alterar, e artigos relacionados à segurança em redes móveis ad hoc têm surgido com uma freqüência maior dentro da comunidade científica.

Pode-se concluir que as questões relativas às redes móveis ad hoc continuam abertas em todos os campos do conhecimento [11] [13], ou seja, ainda não existem soluções que estejam padronizadas para este tipo particular de redes sem infra-estrutura. Este fato mantém as redes em sua embrionária, e também explosiva, fase de desenvolvimento.

Enquanto isso, uma miríade de aplicações ad hoc, com alto potencial de modificar o cotidiano da humanidade, permanece enclausurada no campo das idéias, apenas esperando o desenvolvimento de novos protocolos, modelos e mecanismos adequados a esta nova realidade das redes de computadores para que sejam libertadas. Esta dissertação tem como um dos seus objetivos contribuir com este desenvolvimento, apresentando um mecanismo de segurança capaz de garantir a manutenção da privacidade.

1.3 Escopo

O escopo desta dissertação é a especificação e a implementação do protótipo de um mecanismo de segurança, denominado “Domínios Virtuais para Redes Móveis Ad Hoc: um Mecanismo de Segurança”, que possa

---

[3] *Mobile Ad Hoc Networking Working Group*

garantir alguns aspectos da privacidade entre dispositivos que compõem uma rede móvel ad hoc.

Os aspectos da privacidade garantidos pelo mecanismo estão restritos à privacidade das informações relativas à rede propriamente dita e a aos dispositivos que a compõem, criando condições de impedir que uma parte não-autorizada tenha conhecimento da identidade e dos recursos dos participantes da mesma. O mecanismo proposto nesta dissertação impede ainda que um usuário seja rastreado, através dos dispositivos portáteis que possui, por uma parte não-autorizada, garantindo um anonimato limitado.

O mecanismo proposto deve ser tratado como uma primeira proteção à rede móvel, protegendo-a de atacantes externos[4], ou seja, que a princípio não fazem parte da rede. O mecanismo deve, portanto, ser associado a um ou mais protocolos de autenticação e autorização, capazes de individualizar o controle de acesso aos serviços presentes na rede.

Esta dissertação de mestrado é um resultado acadêmico de um projeto de pesquisa do LARC (Laboratório de Arquitetura e Redes de Computadores) sobre segurança em redes móveis ad hoc, e compreende apenas a uma parte de uma ampla arquitetura de segurança, pertencente a um modelo desenvolvido para estas redes [15] [16].

1.4 Organização do Trabalho

Este item possui, como objetivo, apresentar a organização desta dissertação de mestrado, relacionando e descrevendo brevemente os capítulos que a compõem.

Requisitos e Trabalhos Relacionados

Apresenta uma discussão inicial sobre os aspectos de segurança que devem ser considerados na análise dos requisitos de segurança de uma rede móvel ad hoc, seguida de uma breve explanação sobre os desafios impostos a

---

[4] Atacantes externos se contrapõem a atacantes internos, sendo estes últimos compostos por dispositivos autorizados, porém comprometidos por algum ataque externo, ou usuários registrados mal-intencionados [14].

qualquer mecanismo de segurança a ser desenvolvido para estas redes em particular.

Uma breve discussão sobre fundamentos, problemas e sugestões publicadas sobre questões relativas a aspectos de segurança nas redes móveis ad hoc é feita, com objetivo de contextualizar esta dissertação de mestrado em meio à pesquisa relativa ao tema escolhido, além de ser utilizado como parâmetro para se destacar os aspectos acadêmicos e de inovação tecnológica presentes neste trabalho.

O Mecanismo – Domínios Virtuais

Este capítulo apresenta o projeto do mecanismo de segurança proposto, definindo inicialmente seus requisitos e então apresentando suas funcionalidades.

É dividido em quatro partes principais, na qual a primeira apresenta rapidamente os requisitos funcionais do mecanismo, enquanto que a segunda fornece uma visão geral do mesmo, além de ser, finalmente, nomeado mecanismo de Domínios Virtuais. O item segue com a definição do formato das mensagens trocadas entre os dispositivos móveis e com os fluxogramas que indicam seu funcionamento.

A terceira parte deste capítulo apresenta a arquitetura do mecanismo de Domínios Virtuais, definindo e especificando seus diversos blocos funcionais.

A última parte deste capítulo relaciona o mecanismo de Domínios Virtuais com os demais trabalhos citados no capítulo anterior, de modo a posicioná-lo definitivamente entre os demais, destacando então sua relevância acadêmica.

Implementação

Apresenta os diversos aspectos envolvidos na implementação, assim como algumas características do projeto que estejam mais próximas da implementação, como a máquina de estados do mecanismo de Domínios Virtuais e a definição dos códigos utilizados nos campos das mensagens trocadas.

Este capítulo ainda contém o projeto do mecanismo orientado a objetos, apresentando os casos de uso previstos, além dos diagramas de classes, suas descrições e também os diagramas de interação relacionados.

Testes e Resultados

Este capítulo apresenta os testes realizados e os resultados obtidos a partir destes.

Inicialmente, é apresentada a metodologia a ser utilizada na realização dos testes que, basicamente, divide-os entre os módulos que compõem a aplicação, de modo a avaliá-los separadamente, para que, finalmente, possam ser realizados os testes finais, através da integração de todos os módulos .

Um dos principais itens analisados do mecanismo de Domínios Virtuais é o seu PRNG, que é submetidos a uma série de diferentes testes que incluem, entre outros, o teste de aderência χ^2 e a bateria de testes FIPS PUB[5] 140-2 [17].

Considerações Finais

Apresenta as considerações finais sobre o trabalho, como uma breve discussão sobre os resultados obtidos nesta dissertação frente aos objetivos propostos, e uma visão geral sobre as contribuições deste trabalho. O capítulo segue apresentando propostas sobre a continuação do trabalho apresentado.

---

[5] *Federal Information Processing Standards Publication.*

Capítulo 2

Requisitos e Trabalhos Relacionados

"If you have built castles in the air, your work need not be lost; that is where they should be. Now put the foundations under them."

Henry David Thoreau

Redes móveis ad hoc têm se constituído em um desafio para a comunidade acadêmica envolvida com questões relativas ao desenvolvimento das redes de computadores. A criação de novos paradigmas que consigam atender às necessidades criadas por estas redes singulares tem se concentrado, principalmente, em questões relativas ao roteamento [2], como se pode perceber através do volume de artigos e de propostas de novos protocolos, além da existência de um grupo dentro do IETF [12] destinado exclusivamente à escolha e padronização de um protocolo que, de fato, seja adequado a tais redes.

Protocolos de roteamento adequados são fundamentais para a simples existência de uma rede ad hoc, por se tratar exatamente de um mecanismo que pertence à própria infra-estrutura da rede, o que tem, em parte, justificado a relativa ausência de questões envolvendo a área de segurança do foco da comunidade científica.

No entanto, a segurança em redes móveis ad hoc não pode ser vista apenas como um mecanismo supérfluo, ou uma futura funcionalidade. Mecanismos de segurança devem, de fato, fazer parte das características intrínsecas da própria infra-estrutura da rede móvel ad hoc, assim como de qualquer outra rede de computadores, de modo a garantir a segurança das informações e, sendo assim, devem ser pesquisados e projetados.

2.1 Aspectos de Segurança em Redes Móveis Ad Hoc

Os aspectos relacionados a seguir, também conhecidos como serviços de segurança [18], devem ser considerados na análise da segurança em uma rede móvel ad hoc [14] [19]. São eles:

- Privacidade, de modo a garantir que as informações transmitidas não possam ser lidas ou copiadas por uma parte que não tenha sido autorizada explicitamente pelo proprietário da mesma.
- Disponibilidade, de forma que serviços estejam sempre disponíveis para quaisquer partes autorizadas quando requisitados pelas mesmas, a despeito de eventuais ataques DoS[6].
- Integridade, de modo a garantir que uma mensagem transmitida não seja modificada por falhas ou ataques. Modificações incluem alterações de conteúdo, como adição ou remoção de informação, atraso proposital na transmissão e retransmissão de informação[7] [18].
- Autenticidade, de forma que as partes envolvidas na comunicação possam identificar corretamente a origem de uma dada mensagem. A autenticação é necessária para evitar que ataques de personificação, na qual um usuário não autorizado se passa por um autorizado, possam ser executados.
- Não-repúdio, de modo que a origem ou o destino de uma ação ou mensagem não possa negar a transmissão ou recepção da mesma futuramente.

O controle de acesso, serviço de segurança não citado na relação acima, versa que a capacidade de permitir o acesso a recursos e informações deve ser de responsabilidade do sistema que os oferece, o que é inerente a um

---

[6] *Denial of Service* – uma espécie de ataque, na qual um dispositivo é inundado por requisições de serviço, ocupando tempo de processamento e banda de transmissão, de forma a impedir a disponibilidade do mesmo.

[7] Também conhecido por ataque *replay*, no qual mensagens válidas são retransmitidas com o intuito de se passar por uma parte autorizada, sendo utilizado comumente como mecanismo de ataque contra a autenticação.

dispositivo pertencente a uma rede móvel ad hoc, como explicitado a seguir, no item 2.2.

É importante ressaltar que não existem mecanismos de segurança capazes de assegurar todos os aspectos relacionados acima [18], sendo apenas limitados a alguns serviços de segurança. Deste modo, a manutenção dos serviços de segurança supracitados só pode ser feita através do funcionamento conjunto de diversos mecanismos, que compõem uma arquitetura de segurança.

2.2 Questões de Segurança em Redes Ad Hoc

Alguns desafios, inicialmente levantados como questões a serem resolvidas para a implementação de redes espontâneas [20], podem ser ligeiramente remodelados e então aplicados à segurança em redes móveis ad hoc. Estes desafios são:

- Uma rede móvel ad hoc pode se dividir, juntar-se a outra, desaparecendo e reaparecendo de modo completamente arbitrário, sendo que a simples definição de uma fronteira para esta rede se torna uma questão nebulosa.
- A quantidade de informações que podem ser configuradas previamente a um determinado dispositivo é extremamente limitada, fazendo com que a simples criação de uma lista relacionando dispositivos e serviços confiáveis torne-se uma tarefa impraticável.
- Dispositivos móveis não podem depender de entidades centrais, como servidores, a todo o tempo, pois podem estar afastados destes, sem possibilidade de comunicação. Sendo assim, dispositivos móveis devem ser auto-suficientes em questões relacionadas à segurança.
- Usuários não devem ser obrigados a serem especialistas em configurações de dispositivos, de modo que qualquer intervenção feita por um usuário deve prover uma interface minimamente intuitiva. Como explicitado em [20], usuários são notoriamente péssimos em tarefas de configuração e esta característica é

acentuada em ambientes que possuam requisitos de segurança não triviais, como são as redes móveis ad hoc.

Finalmente, qualquer solução para estes desafios não deve tolher funcionalidades ou características das redes móveis ad hoc de modo que a resposta a um desafio não deve impossibilitar a solução de outro.

2.3 Procura pela Segurança em Redes Ad Hoc

Todo e qualquer artigo ou grupo de pesquisa sobre segurança em redes móveis ad hoc tem, ao menos, uma característica em comum: o fato de ser recente. De fato, os primeiros artigos versando sobre o tema não possuem ainda três anos.

A seguir, é apresentada uma seleção de trabalhos de pesquisa, cada qual acompanhado de suas relativas propostas de mecanismos, arquiteturas e modelos de segurança. Os trabalhos aqui relacionados foram divididos e reorganizados em torno de temas comuns, como um grupo de pesquisa ou um artigo de referência para os demais, e então discutidos e analisados frente às funcionalidades obtidas pelo mecanismo proposto nesta dissertação, apresentadas no item 1.1 e a serem devidamente detalhadas no Capítulo 3.

2.3.1 Modelo de Segurança Resurrecting Duckling

Os artigos relativos ao modelo de segurança *Resurrecting Duckling* [19] [21] fazem parte dos primeiros trabalhos publicados relacionados à segurança em redes móveis ad hoc.

O primeiro artigo da série [19] apresenta o modelo de segurança propriamente dito, acompanhado de alguns dos principais ataques relativos às redes ad hoc, com destaque ao chamado “tortura por privação de sono”[8], nomeado pioneiramente neste artigo.

---

[8] *Sleep depravation torture* - ataque feito à disponibilidade, podendo ser categorizado como um DoS. Neste tipo de ataque o atacante busca o consumo da energia do dispositivo móvel alvo, que é escassa na maioria dos casos, através de uma interação constante, como uma troca de mensagens. O nome se deve ao

Apesar de se denominar um modelo de segurança, alguns conceitos como disponibilidade, integridade e privacidade são tratados pobremente. Isto pode ser observado, por exemplo, no tratamento da privacidade, ao qual argumenta ser de nenhuma importância sem que exista o tratamento prévio do problema de autenticação, na proteção a ataques de "tortura por privação de sono" que é obtida através do uso de baterias reserva, e na restrição do conceito de integridade, associada somente à parte física do dispositivo.

De fato, o cerne do artigo se localiza em seu mecanismo de autenticação. Este apresenta uma abordagem bem humorada para o problema em redes móveis ad hoc, referenciando os dispositivos móveis como patinhos, que nascem espiritualmente através de um processo de registro chamado de encarnação, e morrem espiritualmente deixando o dispositivo à espera de uma nova alma, em um processo chamado de metempsicose[9] reversa. Estas operações são executadas através da utilização de um dispositivo especial, denominado mãe-pato.

O processo de encarnação se daria através do contato físico entre os dispositivos mãe-pato e patinho, de modo a permitir a transferência de um segredo compartilhado. A possível perda de uma chave e, portanto, do controle de um patinho poderia ser resolvida através de um suicídio do patinho, assistido e ordenado por uma terceira e onipotente parte, como o fabricante do dispositivo, que teria poder sobre todos os dispositivos por ele fabricado, o que corresponde a uma infração aos conceitos básicos de uma rede ad hoc, que repudia a utilização de elementos ou entidades centrais.

O primeiro artigo [19] apresenta incompatibilidades com redes móveis ad hoc, pois permite a existência de apenas uma mãe-pato por dispositivo, ou seja, a comunicação segura só existe apenas entre dois dispositivos, a mãe-

---

estado natural dos dispositivos móveis quando não estão em atividade, conhecido por estado de sono (*sleep state*).

[9] **metempsicose**. [Do gr. *metempsychosis*, pelo lat. *metempsychose*] S.f. **1**. *Filos*. Doutrina segundo a qual uma mesma alma pode animar sucessivamente corpos diversos, homens, vegetais ou animais; transmigração.
FERREIRA, Aurélio Buarque de Holanda. Novo Dicionário da Língua Portuguesa. 2ª edição. Rio de Janeiro.

pato e o patinho, não podendo o patinho estabelecer outras, pois cada patinho pode ter apenas uma e conhecida mãe, seguindo o preceito latino "*mater semper certa*"[10].

Na tentativa de contornar tal problema, o segundo artigo da série [21] transforma a mãe-pato em um super-usuário que define políticas de uso através do emprego de credenciais e centraliza o processo de registro em um dispositivo especial, o representante cibernético. A nova solução apresentada, por se basear em um único dispositivo responsável pelo registro e colocar toda a distribuição de regras de confiança sob um único usuário, é centralizadora demais e, portanto, ainda inadequada para uma rede móvel ad hoc.

Os artigos relativos ao modelo de segurança *Resurrecting Duckling*, apesar de possuírem problemas relativos à sua visão centralizadora, muito provavelmente influenciada pelas soluções existentes para as redes infra-estruturadas, são ainda relevantes devido ao pioneirismo na abordagem da segurança em redes móveis ad hoc, e também pela ousada tentativa, um tanto precoce, de se obter um modelo completo para as mesmas.

2.3.2 Entidade Certificadora Distribuída de Zhou e Haas

Zhou e Haas [14] apresentam uma proposta de segurança para redes móveis ad hoc com base na proteção do mecanismo de roteamento e da emissão de certificados.

A disponibilidade é garantida através da proteção do mecanismo de roteamento utilizando-se rotas redundantes, já que informações provenientes de dispositivos comprometidos[11] seriam naturalmente tratadas como inválidas, e estas teriam efeito equivalente a uma informação expirada, comum a redes móveis. Muitos protocolos de roteamento

---

[10] *mater semper certa*, do latim. Ditado romano que prega que a mãe de uma criança é sempre conhecida, o que nem sempre se pode afirmar sobre o pai.

[11] São considerados dispositivos comprometidos todos aqueles que, propositadamente, produzem ou propagam informações de roteamento inválidas, de modo a prejudicar o funcionamento da rede.

propostos, como o ZRP[12] [22], o AODV[13] [25] [26], o TORA[14] [27] e o DSR[15] [28] são capazes de tratar naturalmente múltiplas rotas [14], o que torna a proteção ao roteamento inerente a estes protocolos.

A emissão de certificados, utilizados para assegurar a autenticidade e eventualmente permitir a manutenção da privacidade, corresponde ao ponto mais importante da proposta e é executada por um grupo de dispositivos especiais (n), denominados servidores. Cada servidor possui uma parcela da chave privada utilizada para assinar os certificados requeridos pelos demais dispositivos, sendo necessário um número mínimo de servidores ($t+1$) para assinar um determinado certificado. Este processo é denominado de "criptografia por disparo"[16] [29] [30].

Assume-se que a chave pública do serviço é conhecida por todos os servidores. O processo de assinatura de um certificado é executado por um servidor qualquer que assume a tarefa de combinar os certificados parcialmente assinados pelos demais servidores, que possuem, como apresentado, parcelas da chave privada. Através da posse de ($t+1$) destas parcelas do certificado, um servidor é capaz de emitir um completo e reconhecido pelos demais dispositivos da rede.

A emissão de um certificado por servidores comprometidos fica, portanto, impossibilitada de ocorrer caso estes não sejam em quantidade igual ou superior ao valor mínimo de servidores requeridos. Outra tentativa de corromper o sistema poderia partir da emissão de uma falsa parte da chave privada por um servidor comprometido de modo que uma falsa chave privada seja composta no final do processo. Uma simples verificação da validade da chave gerada pode ser feita frente à chave pública, bastando

---

[12] *Zone Routing Protocol* – as duas partes que compõem o ZRP, o IERP [23] (Interzone Routing Protocol) e o IARP [24] (Intrazone Routing Protocol), são atualmente classificadas como IETF Internet-Draft.
[13] *Ad Hoc On-Demand Distance Vector* – atualmente classificado como IETF Internet-Draft.
[14] *Temporally-Ordered Routing Algorithm*
[15] *Dynamic Source Routing* – atualmente classificado como IETF Internet-Draft.
[16] *Threshold cryptography*

que o dispositivo que combinou as parcelas escolha um novo conjunto das mesmas e gere uma nova chave privada.

A criptografia por disparo, devidamente adaptada às redes móveis ad hoc e proposta por Zhou e Haas, é adequada para redes com uma grande quantidade de nós, permitindo, deste modo, que o número de servidores na rede seja alto e, portanto, que a chave privada possa ser gerada sempre que necessário. Em redes móveis ad hoc com um pequeno número de servidores, estes podem estar separados de tal modo que não sejam suficientes para compor a chave privada do serviço por um longo período de tempo, tornando a solução pouco adequada para redes ad hoc de menor porte na qual os dispositivos possuem alta mobilidade, como redes domésticas.

A proposta de Zhou e Haas faz parte, juntamente com o modelo de segurança *Resurrecting Duckling*, da primeira geração de trabalhos voltados para a segurança em redes móveis ad hoc, fazendo-se constar, quase obrigatoriamente, das referências bibliográficas que envolvem o tema [31] [32].

2.3.3 Projeto Terminodes e Certificados em Cadeia

O projeto de pesquisa Terminodes [33], iniciado no ano 2000 e com final previsto para o ano de 2010, tem como objetivo abordar as principais questões envolvendo redes móveis ad hoc, desde os fundamentos da camada física até as camadas superiores de aplicação [34], incluindo, portanto, os aspectos relacionados à segurança das mesmas [2].

As ameaças à segurança de redes móveis ad hoc são separadas em dois aspectos distintos, são eles [2]:

- Ameaças aos serviços básicos das redes móveis ad hoc, principalmente como ataques aos protocolos de roteamento ou a mecanismos básicos das redes orientadas a serviço.
- Ameaças aos serviços de segurança, como o mecanismo de estabelecimento e gerenciamento de chaves criptográficas simétricas entre dispositivos ou ainda à emissão de certificados.

O artigo [2], apesar de procurar abordar todos os aspectos de segurança relacionados acima, tem, como objetivo, descrever uma proposta de solução para os serviços de segurança, especificamente para a questão de distribuição de certificados, e conseqüentemente de chaves públicas, em uma rede móvel ad hoc.

A proposta considera que os dispositivos da rede móvel ad hoc possam emitir seus próprios certificados, em uma abordagem semelhante à apresentada pelo mecanismo PGP[17] em relação a chaves públicas, sem depender, no entanto, de um diretório de chaves centralizado. Um certificado é emitido por um dispositivo (u) para outro (v) quando o primeiro crê que uma determinada chave pública pertence realmente à segunda parte (v).

Cada dispositivo deve, segundo a proposta, possuir um repositório local contendo alguns certificados selecionados. Estes certificados seriam todos aqueles emitidos pelo dispositivo, somados a alguns certificados selecionados emitidos por outros dispositivos. Assim, no momento em que um dispositivo (u) quiser verificar se uma chave pública pertence mesmo a um determinado dispositivo (v), os repositórios locais são unidos e um grafo de confiança é construído, através da transformação dos certificados presentes nos repositórios em vértices.

O objetivo da construção do grafo é a obtenção de um caminho de certificados entre ambos os dispositivos (u e v), de modo a estabelecer uma relação de confiança por transitividade.

A solução apresentada, no entanto, tem como requisito que todos os usuários sejam honestos, o que significa que a falsificação de certificados inexiste, o que não é uma suposição razoável. A presença de usuários desonestos, ainda segundo o artigo, requer a utilização de alguma métrica de autenticação.

A utilização de princípios de transitividade, comum em tentativas de distribuição de confiança em redes móveis ad hoc [35] [36], deve ser tratada

---

[17] *Pretty Good Privacy*

com cautela, pois é naturalmente insegura caso seja fundamentada unicamente no princípio de transição, ou seja, sem a existência de nenhum outro controle adicional.

2.3.4 Anonimato como Aspecto da Privacidade

O anonimato, ou seja, a proteção de informações relativas à movimentação ou atividades do usuário é uma característica que deve ser garantida em ambientes de computação móvel [37], o que inclui as redes ad hoc. Portanto, o anonimato nada mais é que um aspecto da privacidade, já que este se relaciona à proteção de informações de caráter restrito, ou seja, que não devem ser disponibilizadas a partes não-autorizadas.

Estudos sobre a questão do anonimato em redes de computadores móveis foram inicialmente impulsionados pela possibilidade da utilização de telefones celulares para acesso à rede Internet. A maioria das soluções publicadas, desde então, voltadas para a garantia do anonimato em redes móveis, tem se fundamentado em pseudônimos associados a uma entidade central [37] [38] [39], em uma arquitetura semelhante à encontrada no protocolo MobileIP [40] e, assim sendo, inadequadas para redes móveis ad hoc.

A tecnologia Bluetooth [8] tenta contornar o problema do anonimato, fornecendo aos dispositivos a alternativa de serem encontrados ou não por outros dispositivos, através do controle do serviço de descoberta. Deste modo um dispositivo Bluetooth poderia permanecer invisível a outros dispositivos com o qual não tivesse estabelecido previamente um canal de comunicação mantendo, em teoria, o seu anonimato.

A manutenção do anonimato no Bluetooth pode, no entanto, ser comprometida através de medidas simples, como esperar que o dispositivo alvo inicie uma requisição de serviços, o que tornaria pública sua identidade ou, ainda, explorando características do próprio protocolo, através da identificação de um campo do pacote transmitido, o código de

acesso, que é calculado a partir de parte do endereço físico[18] do dispositivo mestre de uma *piconet*[19] Bluetooth [8] [42].

2.3.5 Questão da Geração e Distribuição de Chaves

A distribuição de chaves em redes móveis ad hoc pode ser uma tarefa problemática devido a algumas características particulares dessas redes, tais como: a eventual impossibilidade de verificação da revogação de um determinado certificado e a possibilidade dos dispositivos pertencerem a diferentes hierarquias de certificados que não possuam nenhuma relação ou acordo de reconhecimento mútuo.

A geração de uma chave de sessão robusta, a partir de um segredo compartilhado, inevitavelmente fraco, é outro objetivo a ser buscado. Existem duas soluções possíveis para a geração de chaves, as contributivas e as não-contributivas.

As soluções não-contributivas são aquelas que dependem unicamente de apenas um dispositivo para gerar a chave de sessão e distribuí-la entre os demais, enquanto que as contributivas são obtidas utilizando o conceito de construção de chaves a partir de informações recebidas dos diversos dispositivos que irão compartilhá-las, sendo estas distribuídas por um dos dispositivos participantes.

Invariavelmente, esta questão tem origem no seguinte cenário: um grupo de pessoas reunidas deseja compartilhar alguma informação, como um arquivo ou uma apresentação, de modo seguro, devendo, portanto, estabelecer uma chave de sessão entre os participantes do encontro.

Este cenário ad hoc deve ser tratado de modo diferente das tradicionais abordagens de segurança, voltadas exclusivamente aos domínios de rede, através da utilização de listas de acesso, por exemplo.

---

[18] Campo LAP (*Lower Address Part*) do BD_ADDR (*Bluetooth Device Address*), sendo que este último é o endereço físico de 48bits do dispositivo Bluetooth. O BD_ADDR é composto por três partes: o já citado LAP, o UAP (*Upper Address Part*) e o NAP (*Non-significant Address Part*) [41].

[19] Unidade básica de uma rede Bluetooth, sendo composta por um dispositivo mestre e até outros sete dispositivos escravos ativos.

Uma nova abordagem foi apresentada por Asokan [43], que propõe a aplicação de um princípio de localidade entre os dispositivos, no qual é possível afirmar que estes compartilham uma mesma informação, que a princípio não pode ser considerada forte o suficiente para ser utilizada como uma chave criptográfica. Através da utilização de uma proposta contributiva, são produzidas e distribuídas chaves simétricas de sessão entre os dispositivos que compartilhavam uma mesma informação.

Outras soluções contributivas para a distribuição de chaves simétricas são estudadas e analisadas em [44].

2.3.6 Outras Propostas para a Questão da Autenticidade

Além das propostas para a solução da questão da autenticidade apresentada nos itens 2.3.1, dentro do modelo de segurança para redes móveis ad hoc Resurrecting Duckling [19] [21], e 2.3.3, dentro do projeto de pesquisa Terminodes [2], outras propostas de autenticação em redes móveis ad hoc foram desenvolvidas por diversos autores. Outras propostas para a questão de autenticação em redes ad hoc são apresentadas, e também brevemente analisados, a seguir.

2.3.6.1 Direcionada a um protocolo de roteamento

Uma proposta para a autenticação de dispositivos móveis, apresentada por Venkatraman e Agrawal [45], e direcionada a um protocolo de roteamento ad hoc específico, apresenta um mecanismo para a autenticação de dispositivos móveis de uma rede que utiliza o protocolo CBRP[20] [46], no qual um dispositivo é, dentro de um contexto geográfico local, eleito pelos demais como sendo o responsável pelo roteamento e, segundo a proposta, também pela autenticação. O protocolo de roteamento CBRP, no entanto, perdeu a condição de Internet-Draft do IETF, ou seja, foi retirado da lista de candidatos a ser o protocolo padrão para redes móveis ad hoc.

Outra proposta direcionada a um protocolo de roteamento específico, o AODV [25] [26], voltada a impedir ataques de personificação, foi

---

[20] *Cluster Based Routing Protocol*

apresentada por Montenegro e Castelluccia [47]. O mecanismo consiste no estabelecimento de um canal seguro de comunicação entre dois dispositivos móveis, através da definição e utilização de um campo no quadro AODV, associado a variantes das chaves públicas dos dispositivos em questão.

2.3.6.2 Fundamentado na confiança

Weimerskirch e Thonet [48] apresentam uma proposta de autenticação de dispositivos para redes móveis ad hoc com baixos requisitos de segurança, na qual as transações executadas possuem um baixo valor associado, e fundamentada na confiança existente entre estes dispositivos. Neste caso, a autenticação pode ser feita através da posse de uma informação comum, como um segredo compartilhado ou a ocorrência de um evento recente, ou ainda através da recomendação feita por outros dispositivos considerados confiáveis.

As recomendações partiriam de dispositivos considerados confiáveis e que já tivessem interagido previamente com o dispositivo em questão, tendo, portanto, algum conhecimento prévio sobre o seu comportamento. As recomendações seriam positivas ou negativas simplesmente, sem nenhum valor agregado a elas.

A recomendação positiva de um dispositivo hostil, feita por um outro dispositivo previamente considerado confiável, poderia causar a perda da confiança, em parte ou em sua totalidade, deste último, sendo esta informação propagada para os demais dispositivos da rede, de modo que estes possam verificar suas relações de confiança. A proposta também prevê a atualização de informação, permitindo assim a alteração do status de um dispositivo, de confiável para não confiável ou vice-versa.

A proposta procura abordar uma solução voltada a uma característica humana, a confiança, na tentativa de resolver a questão da autenticidade em redes móveis ad hoc sem fortes requisitos de segurança. No entanto, a abordagem peca por tratar a transitividade da confiança de forma booleana, por não analisar a questão da tênue fronteira entre os estados de confiança e não-confiança e ainda por não examinar casos específicos relativos à

formação da rede e o estabelecimento inicial de laços de confiança entre os diversos dispositivos da rede móvel.

O modelo de segurança ao qual o mecanismo descrito nesta dissertação é parte componente também utiliza parâmetros de confiança em redes móveis ad hoc para questões de segurança empregando intervalos estatísticos e variáveis estocásticas como medidas de confiança voltado, porém, para a questão do controle de acesso nestas redes [15].

2.3.6.3 Estendendo o modelo Resurrecting Duckling

Balfanz [49] adota a proposta de autenticação do modelo Resurrecting Duckling [19] [21] como um mecanismo de pré-autenticação, o qual executa através de um modo restrito e limitado de comunicação, como contato físico, infravermelho, ou ainda ondas mecânicas, como som ou ultra-som, a troca de certificados, evitando assim que possíveis atacantes possam transmitir informações fraudulentas neste canal de comunicação, em um possível ataque de personificação.

Após o processo de pré-autenticação, a autenticação ocorreria através do uso das chaves públicas trocadas anteriormente e utilizando um protocolo padrão para a definição de uma chave criptográfica de sessão, como, por exemplo, o TLS[21] [50].

Esta proposta, no entanto, por estar fundamentado no modelo Resurrecting Duckling [19] [21], possui muitas limitações inerentes a este modelo, como a necessidade de um método de comunicação à parte para a troca das chaves públicas e a não adequação ao tratamento de redes móveis ad hoc sob um único domínio administrativo, como redes corporativas.

2.3.6.4 Utilizando chaves públicas

A utilização de chaves públicas no processo de autenticação em redes móveis ad hoc tem sido pouco considerada devido a preocupações relativas à limitação da capacidade de processamento e, conseqüentemente, o alto consumo de energia requerido.

---

[21] *Transport Layer Security*

No entanto, a utilização de criptografia de chaves públicas em dispositivos móveis foi realizada por Gupta e Gupta [51] obtendo resultados satisfatórios utilizando uma versão reduzida do TLS, que oprime a verificação do certificado do servidor. Uma rápida análise das atuais tendências tecnológicas, feita no mesmo artigo, indica que a utilização de aceleradores em *hardware* para a realização de operações de criptografia aliada ao crescente aumento da capacidade de processamento dos dispositivos móveis revela que a viabilidade do uso de criptografia utilizando chaves públicas nestes dispositivos é uma questão a ser superada em poucos anos.

Uma arquitetura de segurança para redes móveis ad hoc que utiliza criptografia de chaves públicas foi desenvolvida por Luo e Lu [52]. Esta arquitetura propõe a distribuição da chave privada a ser utilizada na assinatura de certificados, em uma abordagem que se assemelha vagamente à proposta de Zhou e Haas [14]. A chave privada é dividida em parcelas e distribuída entre todos os dispositivos da rede móvel ad hoc, sendo cada uma destas parcelas associada a um valor único de cada dispositivo, como um endereço físico, de modo que a chave só possa ser recuperada na presença de um mínimo k de dispositivos presentes, seguindo o mecanismo proposto por Shamir [53]. A chave pública associada à chave privada é dada como conhecida por todos os dispositivos da rede.

A lista de revogação de certificados, ou apenas CRL[22], é composta por dispositivos suspeitos associados a uma relação de dispositivos acusadores e é compartilhada entre os participantes da rede móvel ad hoc. Para que uma revogação ocorra, deve existir, ao menos, uma relação de k diferentes acusadores para um determinado dispositivo suspeito.

2.3.7 Modelo para Redes Orientadas a Serviços

Uma proposta de arquitetura de segurança para redes móveis ad hoc orientadas a serviços e utilizando chaves públicas foi desenvolvida pelo Laboratório de Arquitetura e Redes de Computadores (LARC), da Escola Politécnica da Universidade de São Paulo [15] [16].

---

[22] *Certificate Revocation List*

A arquitetura proposta assume que a rede móvel é orientada a serviços, ou seja, os participantes da rede podem ser divididos em provedores e usuários de serviços. A autenticação é realizada através da utilização de certificados. Dispositivos selecionados hospedam os chamados Serviços de Registro (SR), capazes de emitir certificados e credenciais de uso de serviços para os demais participantes da rede, criando domínios móveis, e possivelmente não-contínuos, de dispositivos. Os Serviços de Registro podem reconhecer seus pares, de forma a permitir que os dispositivos pertencentes a um determinado domínio possam identificar dispositivos de outros domínios.

Os provedores de serviços definem os requisitos mínimos, ou credenciais, necessários para a execução de uma determinada tarefa, sendo o controle de acesso executado individualmente. Do mesmo modo, a CRL possui informações locais, ou seja, restrições definidas localmente pelo serviço, além de uma parte comum, sendo então consolidada por um dos Serviços de Registro da rede. Uma vez finalizada, a CRL é propagada e a distribuída entre os diversos dispositivos que compõem a rede. Um controle de versões é ainda implementado, de modo que apenas atualizações da CRL são transmitidas, diminuindo o tráfego da rede.

O mecanismo a ser descrito nesta dissertação de mestrado é parte componente desta arquitetura de segurança.

2.3.8 Considerações

Os diversos trabalhos apresentados no decorrer do item 2.3 procuraram situar o atual estado da questão de segurança em redes móveis ad hoc.

O objetivo maior deste capítulo, no entanto, não é a mera apresentação de trabalhos e artigos, mas sim permitir a criação de condições para que o mecanismo possa ser compreendido dentro de seu contexto de pesquisa, justificando assim o seu desenvolvimento e a sua aplicação.

A seleção dos trabalhos descrito neste capítulo só será devidamente compreendida de fato ao final do item 3.4, no qual o mecanismo proposto nesta dissertação, a ser especificado ao longo do próximo capítulo, será colocado junto aos demais trabalhos, e as relações serão apresentadas.

Capítulo 3

O Mecanismo - Domínios Virtuais

"Once you eliminate the impossible, whatever remains, no matter how improbable, must be the truth."

Sir Arthur Conan Doyle

A descrição de um recurso capaz de oferecer um primeiro passo na direção de um ambiente ad hoc seguro, criando assim condições para que outras aplicações e mecanismos de segurança possam ser aplicados é o objetivo deste capítulo, relativo ao projeto. Assim sendo, são apresentados neste capítulo:

- Os requisitos funcionais do mecanismo de segurança a ser definido e projetado, obedecendo às premissas impostas nos capítulos anteriores.
- A especificação de um mecanismo de segurança que permite que um grupo de dispositivos possa iniciar uma comunicação com condições de assegurar que os requisitos de segurança impostos sejam obedecidos.
- A definição da arquitetura do mecanismo proposto, ou seja, a apresentação detalhada de seus diversos blocos funcionais e da relação entre os mesmos. Esta apresentação é acompanhada, quando necessário, de uma análise sucinta orientada aos requisitos definidos previamente no item 2.2.
- A contextualização do mecanismo, posicionando-o definitivamente entre os demais trabalhos publicados, e apresentados no item 2.3, finaliza o capítulo. Neste item é descrita a integração do mecanismo de Domínios Virtuais com os demais trabalhos, assim como as vantagens e os ganhos obtidos através da utilização conjunta destes.

3.1 Requisitos Funcionais

A partir do levantamento dos aspectos e questões de segurança em redes móveis, apresentados nos itens 2.1 e 2.2 do capítulo anterior, associadas às restrições descritas no item 1.3, é possível determinar, quando colocadas frente à descrição dos objetivos a serem atingidos por esta dissertação de mestrado, conforme apresentado no item 1.1, os requisitos funcionais do mecanismo.

Deste modo, dentre os aspectos de segurança apresentados no item 2.1, a privacidade é destacada como relevante para o mecanismo proposto, pois a proteção das informações relativas à rede de partes não autorizadas pode ser garantida através da manutenção deste serviço de segurança específico. Do mesmo modo, a ocultação da movimentação dos dispositivos da rede de partes não-autorizadas está relacionada à manutenção da privacidade destas informações.

Igualmente relevantes são todas as questões de segurança relativas às redes ad hoc apresentadas no item 2.2. Assim sendo, o mecanismo deve respeitar as restrições impostas pelo ambiente ad hoc, sem tolher funcionalidades e características das mesmas, solucionando questões como a má definição das fronteiras da rede móvel e a independência de entidades centrais, entre outras levantadas no mesmo item.

3.2 Domínios de Redes Virtuais

O mecanismo proposto nesta dissertação de mestrado atende os requisitos previamente definidos no item anterior, 3.1, através da criação de domínios virtuais compostos por dispositivos considerados previamente confiáveis, definindo, deste modo, uma fronteira entre estes elementos e os demais dispositivos da rede de computadores.

Dentro do escopo desta dissertação de mestrado, um domínio virtual corresponde a uma rede móvel ad hoc, na qual os dispositivos participantes desta são considerados confiáveis.

Os domínios virtuais propostos não são geograficamente definidos ou limitados e também não são centralizados em um único dispositivo da rede. A fronteira do domínio virtual, portanto, não é necessariamente contínua, podendo esta se dividir e se unir de acordo com a movimentação de seus dispositivos.

3.2.1 Endereçamento dos Domínios Virtuais

Um domínio virtual é composto por dispositivos que possuem o mesmo endereço de domínio virtual. Este endereço é uma informação restrita aos seus participantes e é, portanto, considerada confidencial. Sendo assim, para que um dispositivo pertencente a um domínio virtual possa reconhecer e ser reconhecido por seus pares, ele deve verificar se as outras partes envolvidas na comunicação possuem o mesmo endereço de domínio virtual.

Como as mensagens trocadas entre as partes envolvidas neste processo de autenticação de grupo podem ser facilmente monitoradas por outras partes não-autorizadas, caracterizando um ataque passivo, que é de difícil detecção em ambientes móveis, outras medidas de segurança são necessárias para evitar que possam ocorrer subseqüentes ataques de repetição, utilizando informações obtidas passivamente. Contramedidas comuns a este tipo de ataque envolvem a utilização de associações com o tempo ou a utilização de desafios [18].

3.2.2 Funcionamento do Mecanismo

A verificação de que ambas as partes são pertencentes a um mesmo domínio virtual é feita através da utilização da troca de mensagens previamente conhecidas, em um processo de desafio fracamente associado ao tempo.

Assim, a parte que deseja se comunicar apresenta inicialmente uma mensagem pré-definida para uma segunda parte, que deve responder com uma segunda mensagem, também de comum conhecimento a ambas as partes, juntamente com a exigência de uma contraprova que deve ser enviada pela primeira parte, de forma a garantir à segunda parte que a

primeira possui realmente informações necessárias para iniciar uma comunicação.

A informação trocada entre ambas as partes corresponde ao próprio endereço do domínio virtual que, devido às suas características previamente apresentadas ainda neste item, não é constante, variando com o passar do tempo.

Deste modo, o endereço do domínio virtual deve ser imprevisível para partes não-autorizadas. Assim sendo, os dispositivos pertencentes a um domínio virtual mantêm a existência do mesmo oculta a partes não-autorizadas, pois não respondem a requisições feitas pelas últimas, já que estas não possuem o conhecimento do endereço do domínio virtual.

Para isto o endereço do domínio virtual é calculado através da utilização de um gerador pseudo-aleatório que possui o tempo atual como um de seus parâmetros de entrada. O gerador pseudo-aleatório utilizado também é imprevisível tanto para a direita como para a esquerda, correspondendo, portanto, a um gerador criptográfico seguro [54].

3.2.2.1 A inspiração

O mecanismo proposto nesta dissertação foi inspirado no modo de operação dos pulos de freqüências, utilizada no mecanismo de espalhamento espectral. Este método, desenvolvido durante a segunda a guerra mundial, exige que as partes que queiram se comunicar compartilhem uma mesma informação, que é a ordem, ou o caminho percorrido, dos pulos de freqüências [55] [56].

Este método, no entanto, não é correlacionado com o tempo, pois não possui um instante de início de transmissão, devendo a parte receptora procurar, ou esperar, por um determinado instante na seqüência de pulos. Assim sendo, a simples aplicação de um mecanismo analogamente semelhante ao espalhamento espectral por pulos de freqüência na tentativa de se obter um ambiente computacionalmente seguro não seria suficiente, exigindo, portanto, modificações e alterações, como a inclusão de um fator temporal que mantivesse a seqüência de autenticação alterada constantemente, e protegida de ataques de repetição.

3.2.2.2 Um mecanismo semelhante – RSA SecurID

Um mecanismo semelhante ao proposto nesta dissertação de mestrado é a solução de autenticação SecurID da RSA Security Inc. Essa solução utiliza pequenos dispositivos portáteis que, assim como o mecanismo de Domínios Virtuais, faz uso de um gerador de números pseudo-aleatórios, que possui como parâmetros de entrada uma semente compartilhada e uma informação de tempo. Deste modo, um novo valor é gerado e apresentado a cada minuto ao usuário, que deve digitá-lo juntamente com sua senha pessoal no momento da autenticação [57].

No entanto, o SecurID não é adequado para uma rede móvel ad hoc, já que sua solução depende de um repositório central para o armazenamento de sementes, além de exigir o sincronismo entre os relógios dos dispositivos portáteis e do servidor. O mecanismo de Domínios Virtuais, ao contrário, independe de entidades centrais, utilizando sementes distribuídas e não exige o sincronismo dos relógios dos dispositivos que compõem um determinado domínio de dispositivos.

3.2.2.3 Seqüência de troca das mensagens

A seqüência de mensagens trocadas entre as partes envolvidas na comunicação é apresentada na Figura 3.1, abaixo, que apresenta dois dispositivos, **A** e **B**, pertencentes ao mesmo domínio virtual em um processo de identificação das partes.

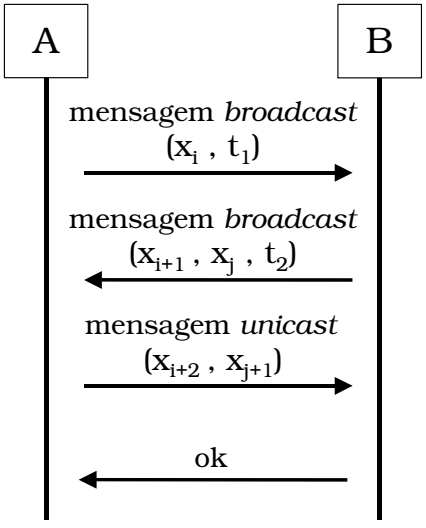

Figura 3.1: Reconhecimento de um domínio virtual.

Na Figura 3.1, acima, inicialmente o dispositivo **A,** também chamado de requisitante, procura por outros dispositivos que também pertençam ao seu domínio virtual. Isto é feito através do envio de uma mensagem de desafio inicial *broadcast*, ou seja, a todos os demais dispositivos na área de alcance, contendo um elemento, x_i, gerado a partir de uma seqüência pseudo-aleatória associada ao valor temporal t_1. Esta mensagem é enviada com o campo de endereço origem vazio, de modo que o dispositivo **A** não possa ser identificado a princípio.

Todos dispositivos que receberem a mensagem proveniente de **A** devem verificar se x_i corresponde a um valor esperado de sua seqüência pseudo-aleatória associada ao valor temporal t_1. Os dispositivos que receberam esta mensagem podem então apresentar dois comportamentos distintos:

- Caso o dispositivo que tenha recebido a mensagem, **B**, verifique que x_i não pertence ao conjunto de valores esperados, este permanece em silêncio, de modo a não ter sua presença revelada ao dispositivo requisitante.
- Caso o dispositivo que tenha recebido a mensagem, **B**, verifique que x_i realmente pertence ao conjunto de valores esperados, este responde com uma resposta *broadcast*, contendo o elemento seguinte da série, x_{i+1}, que corresponde à contraprova do desafio colocado por **A**, além de outro elemento, x_j, igualmente gerado em uma seqüência pseudo-aleatória associada, porém, a um segundo valor temporal t_2. Sendo assim, o dispositivo requisitante **A** pode apresentar dois comportamentos distintos:
  - ❖ Caso o dispositivo requisitante **A** verifique que o valor x_{i+1}, recebido do segundo dispositivo, **B**, não corresponde ao próximo valor da seqüência pseudo-aleatória iniciada x_i, o processo é finalizado e reiniciado. O segundo dispositivo passa, então, a ser ignorado pelo requisitante em futuras tentativas por algum tempo.
  - ❖ Caso o dispositivo requisitante **A** verifique que o valor x_{i+1}, recebido do segundo dispositivo, **B**, corresponde ao próximo

valor da seqüência pseudo-aleatória iniciada x_i, e que o segundo valor recebido, x_j, corresponde ao valor obtido pelo gerador pseudo-aleatório quando associado ao valor temporal t_2, o dispositivo requisitante deve enviar a sua contraprova, ou *réplica*, para o segundo dispositivo em uma mensagem *unicast*, ou seja, endereçada apenas para um dispositivo, no caso, o requisitante, que contenha os valores que seguem os últimos recebidos, ou seja, x_{i+2} e x_{j+1}. Estes valores são então verificados pelo segundo dispositivo, e caso corretos, **B** assume que **A** pertence ao seu domínio de rede virtual, caso contrário, encerra a comunicação, assumindo que **A** não pertence ao seu domínio.

3.2.2.4 Posicionamento do mecanismo

O mecanismo de Domínios Virtuais encontra-se posicionado entre a aplicação e o bloco de comunicação de dados, como apresentado na Figura 3.2, abaixo.

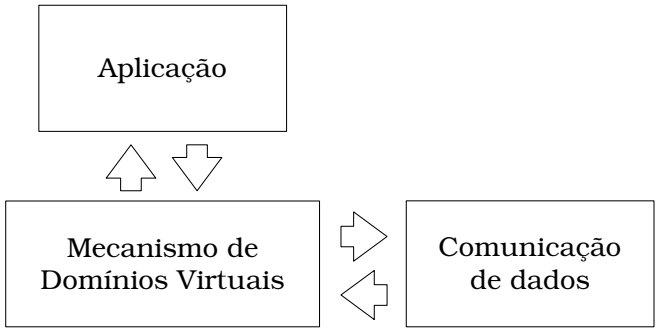

Figura 3.2: Posicionamento do mecanismo de Domínios Virtuais.

A aplicação corresponde a um processo que demanda a utilização de um serviço de comunicação enquanto que o bloco de comunicação de dados é responsável pela troca de dados entre dispositivos utilizando uma tecnologia de redes de computadores sem fio qualquer[23], como o Bluetooth [8] ou o IEEE 802.11 [6] [7].

[23] A utilização de uma rede de computadores sem fio não se trata de uma limitação do projeto, mas a uma escolha determinada pela utilização de redes móveis ad hoc.

3.2.2.5 Formato das mensagens

Uma vez definido o conteúdo de cada uma das mensagens a serem transmitidas pelo mecanismo de Domínios Virtuais é possível determinar um formato para as mesmas. O campo correspondente ao valor aleatório possui tamanho variável, já que é dependente do gerador de números pseudo-randômicos a ser utilizado pelo mecanismo. Os demais campos também não têm seus tamanhos determinados definitivamente a princípio, pois dependem dos tipos de endereço a serem utilizados, assim como da representação do índice de tempo a ser aplicada.

Deste modo, todas as mensagens utilizadas pelo mecanismo de Domínios Virtuais podem ser representadas a partir da configuração apresentada na Figura 3.3,abaixo.

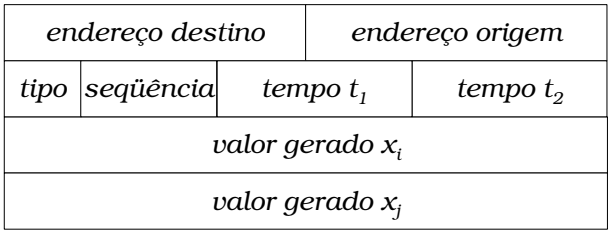

Figura 3.3: Formato de mensagem padrão utilizada pelo mecanismo.

Sendo que o campo *tipo* determina o tipo de mensagem enviada, ou seja, se a mensagem em questão é um *desafio*, uma *resposta* ou uma *réplica* e o campo *seqüência* é utilizado para associar réplicas a respostas e estas a desafios.

Utilizando a mensagem padrão definida acima, é possível determinar o conteúdo do campo de cada uma das mensagens utilizadas pelo mecanismo de Domínios Virtuais. Assim sendo, uma mensagem do tipo *desafio*, enviada por um dispositivo *A*, apresenta endereço destino do tipo *broadcast*, campo seqüência contendo um valor qualquer *n*, tempo t_1 e um valor gerado x_i. Os campos endereço origem, tempo t_2 e valor gerado x_j são nulos, como apresentado na Figura 3.4.

O mecanismo é, no entanto, independente de tecnologia de comunicação, podendo ser aplicado inclusive em redes não-móveis.

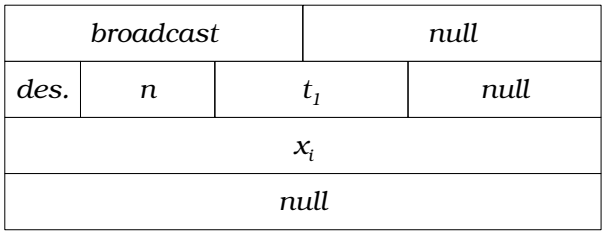

Figura 3.4: Mensagem *desafio*.

Uma mensagem do tipo *resposta*, enviada pelo dispositivo *B*, tem como endereço destino do tipo *broadcast*, já que não é possível determinar o endereço origem da mensagem *desafio*. O campo endereço de origem é preenchido com o próprio endereço do dispositivo *B*, enquanto que o campo de seqüência apresenta o valor *n+1*. O tempo t_1 permanece inalterado, enquanto que t_2, um segundo parâmetro de tempo, é acrescentado. A mensagem é completada com o acréscimo do próximo valor da seqüência enviada por *A*, x_{i+1}, e pelo valor x_j, relacionado ao valor t_2. A estrutura da mensagem do tipo *resposta* é apresentada na Figura 3.5, abaixo.

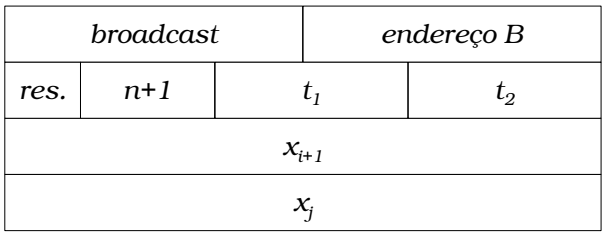

Figura 3.5: Mensagem *resposta*.

A última mensagem prevista no mecanismo é do tipo *réplica*, a ser enviada pelo dispositivo *A*, e possui como endereço destino o dispositivo *B*, enquanto que o campo correspondente ao endereço de origem é preenchido com o próprio endereço do dispositivo *A*. O campo de seqüência apresenta agora o valor *n+2* e os parâmetros de tempo t_1 e t_2 permanecem inalterados. Os campos restantes da mensagem correspondem ao valor seguinte da seqüência iniciada por *A*, x_{i+2}, e pelo valor seguinte da seqüência enviada por *B*, x_{j+1}, relacionado ao valor t_2. A estrutura da mensagem do tipo *réplica* é apresentada na Figura 3.6.

endereço B		*endereço A*	
rep.	*n+2*	t_1	t_2
x_{i+2}			
x_{j+1}			

Figura 3.6: Mensagem *réplica*.

3.2.2.6 Fluxogramas

Os fluxogramas que representam a seqüência de troca de mensagens descrita no item anterior, 3.2.2, tanto do dispositivo requisitante **A** como dos demais dispositivos da rede, representados por **B**, são, a seguir, apresentados.

A Figura 3.7 contém o fluxograma do mecanismo durante a procura de um determinado domínio virtual. Como apresentado nesta figura, a parte inicial da seqüência é calculada e transmitida, levando o dispositivo ao estado de espera da resposta da primeira mensagem. Assim que a resposta é recebida, a última parte do desafio é então calculada e enviada, assim como é calculada e enviada a resposta a um segundo desafio, enviado pelo segundo dispositivo. Assim que a confirmação do processo é recebida, o processo pode ser completado, caracterizando o reconhecimento de um dispositivo pertencente ao mesmo domínio virtual do requisitante.

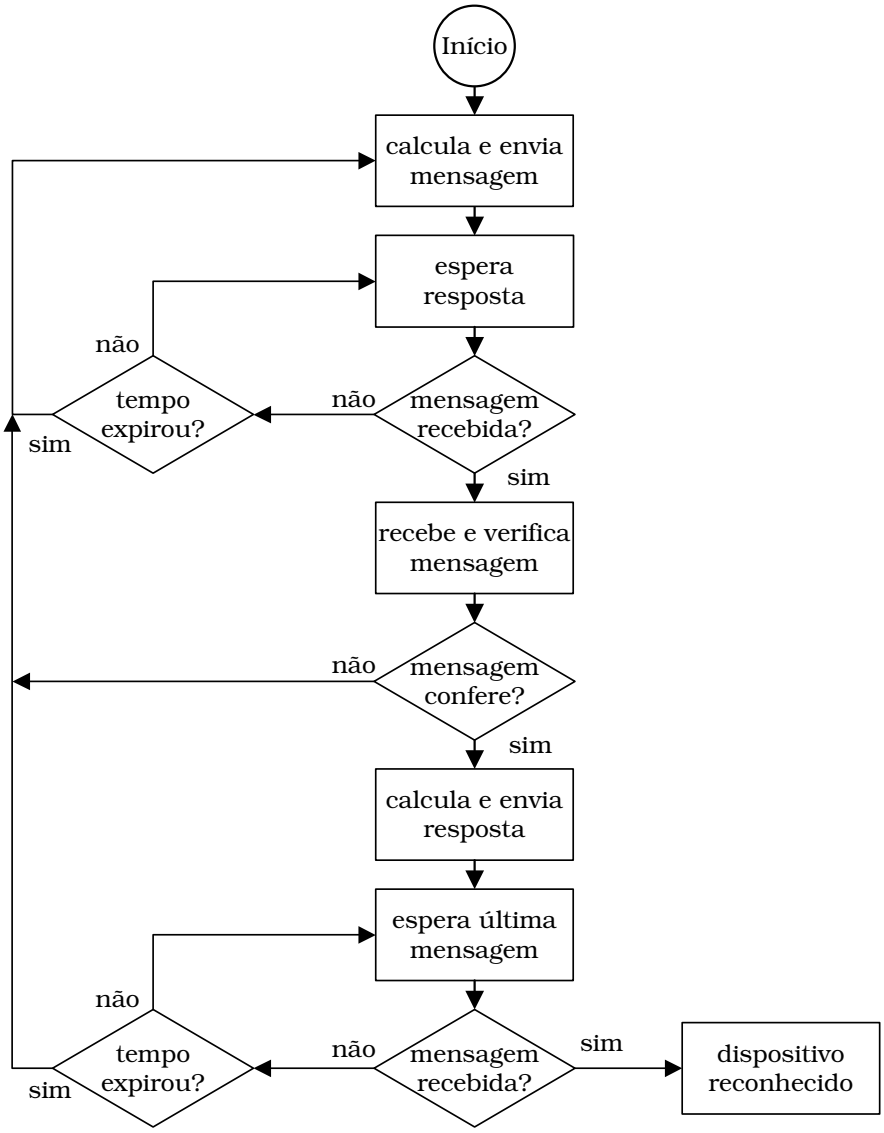

Figura 3.7: Fluxograma da procura por um domínio virtual.

O mecanismo pode, naturalmente, ser interrompido a qualquer instante, caracterizando a desistência da procura pelo domínio virtual.

A Figura 3.8 apresenta o fluxograma do mecanismo durante o processo de espera e eventual reconhecimento e de participantes de um determinado domínio virtual. Como apresentado na figura, inicialmente, o dispositivo espera por mensagens que reconheça. Assim, toda mensagem recebida que se caracterize como sendo uma procura por um domínio virtual é verificada e, caso não seja reconhecida, é simplesmente ignorada. No entanto, caso seja reconhecida, a segunda parte da seqüência é calculada, assim como uma nova seqüência de desafio, associada a um segundo valor temporal.

Uma mensagem é construída a partir destes dados, que são então transmitidos, levando o dispositivo a um novo estado de espera pela próxima e última mensagem.

Assim que esta resposta é recebida, a última parte do desafio é então enviada. O recebimento da última mensagem esperada indica que o processo pode ser completado, caracterizando o reconhecimento de outro dispositivo pertencente ao mesmo domínio virtual do requisitante. Uma mensagem final é então transmitida, de modo a sinalizar ao dispositivo requisitante que o procedimento foi bem sucedido.

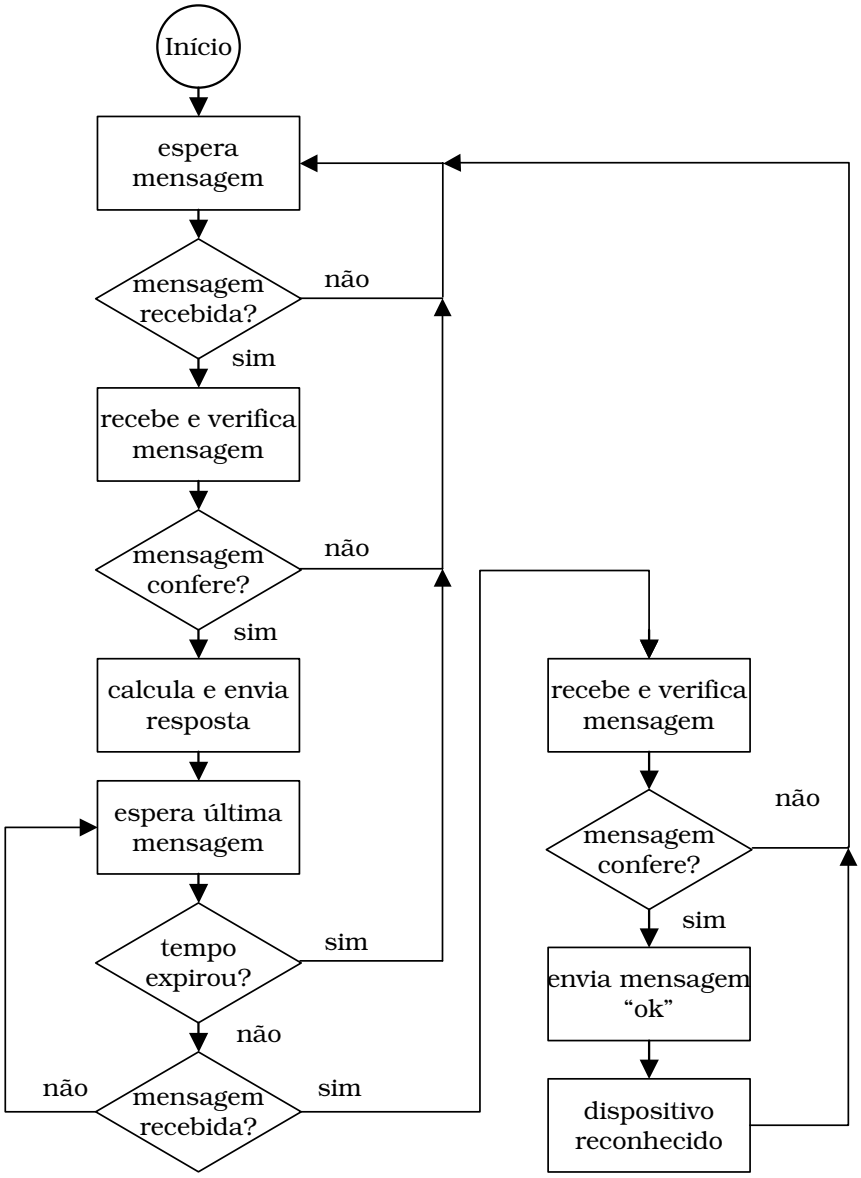

Figura 3.8: Fluxograma da requisição de um domínio virtual.

O processo apresentado acima é capaz de identificar os participantes de um domínio de rede virtual através da utilização de provas e contraprovas produzidas através de um gerador pseudo-aleatório imprevisível, tanto para a direita como para a esquerda, na ausência de uma informação privilegiada, como o conhecimento da semente utilizada no gerador.

3.2.2.7 Considerações sobre a troca das mensagens

Algumas considerações devem ser feitas sobre a troca de mensagens entre as duas entidades apresentada no item anterior, 3.2.2.3, no que diz respeito às seqüências de números pseudo-aleatórios geradas, à sua distribuição e suas características de segurança, como a impossibilidade da dedução de uma seqüência a partir de outras ou ainda a proteção contra ataques de repetição. Estas considerações, relativas aos geradores de números pseudo-aleatórios, ou simplesmente PRNG[24], são discutidas a seguir, no item 3.3.

3.3 Arquitetura do Mecanismo de Domínios Virtuais

A arquitetura do mecanismo de Domínios Virtuais é constituída pelos seus blocos funcionais, que são apresentados na Figura 3.9.

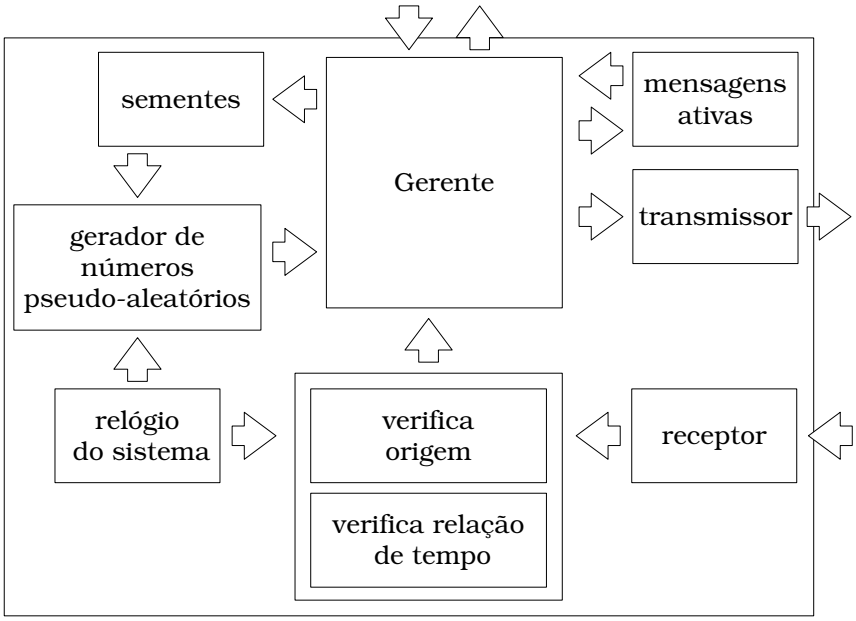

Figura 3.9: Arquitetura do mecanismo de formação de Domínios Virtuais.

[24] *Pseudo-random number generator*

Os blocos funcionais da arquitetura são relacionados e, brevemente, descritos abaixo:

- Um *gerador de números pseudo-aleatórios*, cujo papel é definir o endereço do domínio virtual, através da construção das seqüências de desafios e respostas.
- *Sementes*, sendo que cada uma corresponde a um diferente domínio virtual.
- *Relógio do sistema*, utilizado na atribuição da característica temporal ao sistema.
- Os blocos de *transmissão e recepção* de dados, utilizados para a comunicação do mecanismo com o meio externo.
- Um bloco de *verificação da relação temporal*, que analisa os desafios recebidos, mantém uma relação de valores de tempo utilizados em desafios anteriores e recebe os pedidos de comunicação.
- Um bloco de *verificação do endereço origem* que analisa as respostas a desafios enviados, de forma a detectar e filtrar ataques ou dispositivos atacantes.
- Um *gerente*, responsável pela comunicação do mecanismo com o sistema computacional, pela seleção da semente a ser utilizada na geração dos números pseudo-aleatórios, pela verificação dos valores pseudo-aleatórios recebidos e também pelo controle do estado de mensagens enviadas e recebidas pelo mecanismo.
- Um bloco contendo as *mensagens ativas*, ou seja, uma relação de mensagens enviadas e recebidas e o seu status corrente.

Os itens a seguir apresentam detalhadamente cada um dos blocos funcionais que compõem a arquitetura do mecanismo de Domínios Virtuais.

3.3.1 Gerador de Números Pseudo-aleatórios

O PRNG é responsável pela construção dos desafios utilizados pelo mecanismo de Domínios Virtuais. Um gerador de números pseudo-aleatórios (PRNG) nada mais é que um algoritmo determinístico que produz seqüências de números aparentemente randômicos a partir de um valor inicial, ou semente [54] [58].

O mecanismo de Domínios Virtuais proposto utiliza um PRNG com propriedades criptográficas, ou seja, um observador externo que possua conhecimento histórico dos valores gerados previamente não deve ser capaz de inferir qual o próximo valor a ser produzido em uma seqüência. Sendo assim, uma seqüência gerada através da utilização de um PRNG que possua propriedades criptográficas não deve ser distinguida de uma seqüência randômica de valores [59] [60].

Além de obedecer aos requisitos relativos à aleatoriedade e imprevisibilidade dos valores gerados, o PRNG deve também ser capaz utilizar um ou mais fatores temporais como parâmetros de entrada, permitindo a geração de uma mesma seqüência de valores em dois dispositivos que compartilhem um mesmo segredo.

Os principais PRNG com propriedades criptográficas, estudados e analisados em [61] segundo suas características de segurança, juntamente com os geradores de seqüências Blum Blum Shub [62] [63] e o recente Yarrow-160 [60], são considerados e avaliados nos itens a seguir, frente às necessidades impostas pelos requisitos do mecanismo de Domínios Virtuais. Modificações são sugeridas e apresentadas quando necessário, de forma a adequá-los ao mecanismo proposto.

3.3.1.1 PRNG ANSI X9.31

O PRNG ANSI[25] X9.31 é o gerador utilizado pelo algoritmo de criptografia simétrica DES[26] para produzir suas chaves e faz uso de outro algoritmo de

---

[25] *American National Standards Institute*
[26] *Data Encryption Standard*

criptografia simétrica, tendo o 3DES[27], como bloco fundamental de funcionamento[28]. O funcionamento do algoritmo é o seguinte [18] [61] [64]:

$$saída[i] = E_K(T_i \oplus semente[i]), \text{ sendo que } T_i = E_K(timestamp) \quad (1)$$

$$semente[i+1] = E_K(T_i \oplus saída[i+1]) \quad (2)$$

Onde *E* é o algoritmo de criptografia de blocos 3DES, *K* é a chave secreta utilizada pelo 3DES e que nunca é alterada pelo algoritmo e *timestamp* é uma entrada que representa o valor de tempo corrente.

De modo a atender aos requisitos do mecanismo de Domínios Virtuais, algumas mudanças devem ser efetuadas no algoritmo, como a eliminação do passo de atualização da semente, equação (2), já que esta deve permanecer constante ao longo do tempo, já que a atualização desta semente em todos os dispositivos da rede móvel ad hoc não seria viável. Assim sendo, o algoritmo resultante adaptado aos requisitos impostos ficaria da seguinte forma:

$$saída = E_K(T_i \oplus semente), \text{ sendo que } T_i = E_K(timestamp) \quad (3)$$

Onde a chave secreta *K* é também a própria semente do gerador.

Este gerador apresenta, no entanto, uma limitação. Para um determinado valor de tempo é possível gerar apenas 64bits de informação, o que corresponde à saída do algoritmo de criptografia do 3DES, a serem divididos em três mensagens.

Isto acarreta a utilização de uma mensagem inicial de desafio e uma resposta de 30bits com 4bits de réplica, de modo que a possibilidade de um atacante enviar uma resposta correta a um desafio escolhendo aleatoriamente um valor dentro deste universo é de cerca de 10^{-9}. Utilizando um desafio e uma resposta de 32bits e supressão da réplica, a possibilidade é de cerca de 2.10^{-10}. No entanto, um atacante poderia enviar múltiplas

---

[27] *Triple DES (Data Encryption Standard)*

[28] O 3DES pode, no entanto, ser substituído por qualquer outro algoritmo de criptografia de blocos [18].

respostas para este desafio, desde que dentro da janela de tempo que determina a validade do mesmo (ver item 3.3.2), o que aumentaria a possibilidade de sucesso de um ataque.

Uma alternativa à limitação do uso de 64bits seria o uso de dois ou mais *timestamps*, de modo a se ter ao menos 128bits a serem divididos entre as mensagens de desafio, resposta e réplica.

3.3.1.2 PRNG DSA

O PRNG DSA[29] é um algoritmo utilizado para gerar números randômicos utilizando a função de *hash* SHA-1[30] e pertence ao padrão DSS[31] da NIST[32], que especifica um conjunto de algoritmos que podem ser utilizados para produzir uma assinatura digital [65]. Abaixo, o algoritmo é apresentado:

$$saída[i] = hash\left((W_i + X_i) \bmod 2^b\right), \text{ com } 160 \le b \le 512 \quad (4)$$

$$X_{i+1} = (1 + X_i + saída[i]) \bmod 2^b \quad (5)$$

Onde *Wi* corresponde a um valor opcional definido pelo usuário e *Xi* é um valor secreto correspondendo à semente do gerador.

De modo a adequar este algoritmo às necessidades do mecanismo de Domínios Virtuais, deve-se retirar o passo de atualização da semente, equação (5), já que ela deve permanecer constante ao longo do tempo, e utilizar *Wi* como parâmetro temporal do algoritmo. Assim sendo, o PRNG fica:

$$saída = hash\left((timestamp + semente) \bmod 2^b\right), \text{ com } 160 \le b \le 512 \quad (6)$$

Uma execução deste gerador oferece um número aleatório de 160bits de comprimento, o que possibilita a utilização de mensagem de desafio e resposta de 64bits, e ainda uma réplica de 32bits, de modo que a possibilidade de sucesso de um possível atacante responder corretamente a

---

[29] *Digital Signature Algorithm*
[30] *Secure Hash Algorithm*
[31] *Digital Signature Standard*
[32] *National Institute of Standards and Technology*

um desafio escolhendo aleatoriamente um valor dentro do universo de possibilidades é de aproximadamente 5.10^{-20}, e de cerca de 10^{-29} para enviar o desafio e a réplica esperados consecutivamente em um determinado instante. Novamente, deve-se ressaltar que um atacante pode enviar múltiplas respostas a um desafio dentro da janela de tempo que determina a validade do mesmo (ver item 3.3.2), o que aumentaria a possibilidade de sucesso de um ataque.

A supressão de alguns bits do valor randômico produzido também pode ser feita através de uma réplica de apenas 8bits, por exemplo, de modo a ocultar parte do valor aleatório, 24bits no exemplo, e assim dificultar um eventual processo de criptoanálise dos valores gerados pelo PRNG.

3.3.1.3 PRNG RSAREF

O RSAREF v.2.0 é uma implementação de referência de criptografia proposta pelo RSA Laboratories e apresenta um PRNG que utiliza a função de *hash* MD5 e adições módulo 2^{128} [61] [66]. O algoritmo, apresentado a seguir, está dividido em duas partes, sendo a primeira relativa à entrada de informação no gerador, equação (7), ou seja, a utilização de sementes e a segunda relativa à saída de informação, equações (8) e (9), ou seja, dos valores pseudo-randômicos produzidos.

A equação (7), abaixo, apresenta o método de tratamento de índices de entrada.

$$C_{i+1} = \left(C_i + hash\left(semente_i\right)\right) \bmod 2^{128} \tag{7}$$

As equações (8) e (9), a seguir, apresentam o método de geração de valores randômicos.

$$saída[i] = hash\left(C_i\right) \bmod 2^{128} \tag{8}$$

$$C_{i+1} = \left(C_i + 1\right) \bmod 2^{128} \tag{9}$$

Onde *Ci* corresponde a um contador de 128bits.

O algoritmo, no entanto, não pode ser adaptado para sua utilização no mecanismo de Domínios Virtuais sem que haja a completa descaracterização do RSAREF PRNG, pois este faz uso de um contador de 128bits que é incrementado a cada valor gerado, o que impossibilita o uso do mesmo em um grupo de usuários que provavelmente não estará sempre próximo e, portanto, não seria possível contar com a atualização do contador em todos os dispositivos.

3.3.1.4 PRNG Blum Blum Shub

O PRNG Blum Blum Shub é um gerador seguro do ponto de vista criptográfico por ser imprevisível tanto para a direita como para a esquerda, ou seja, dado um trecho de uma seqüência de valores gerados não é possível determinar o elemento que se encontra a esquerda ou a direita deste trecho com a probabilidade maior que o um dividido pelo universo de respostas possíveis [62] [63]. O algoritmo utilizado, baseado em resíduos quadráticos, está apresentado abaixo [59]:

$$S_{i+1} = \left(S_i^{\,2}\right)\left(\mathrm{mod}\, n\right), \text{ sendo que } S_0 = \left(x^2\right)\left(\mathrm{mod}\, n\right) \tag{10}$$

Onde *n* é um valor determinado pela multiplicação de dois números primos, *p* e *q*, sendo que ambos apresentam resto três quando divididos por quatro, e *x* é primo relativo a *n*. Outro ponto importante deste PRNG é que para que sua característica de segurança seja garantida, deve-se utilizar não mais que *k* bits da ordem menos significativa do valor produzido em uma interação, onde *k* é:

$$k = \log_2\left(\log_2\left(S_i\right)\right) \tag{11}$$

O PRNG Blum Blum Shub é adequado aos requisitos funcionais do mecanismo de Domínios Virtuais, não existindo a necessidade de qualquer alteração no seu algoritmo, já que o valor *n* pode ser público, desde que *x* seja secreto [59]. Esta propriedade é adequada para a transmissão do valor de tempo, que deve ser público.

No entanto, o PRNG Blum Blum Shub é computacionalmente intensivo [59], o que o torna lento quando comparado com os demais geradores

apresentados, além de exigir uma maior quantidade de processamento aritmético e conseqüente gasto de energia, o que é uma restrição no caso de dispositivos móveis, já que esta é, invariavelmente, um recurso extremamente limitado.

3.3.1.5 PRNG Yarrow-160

O PRNG Yarrow-160 foi projetado focando potenciais ataques sobre o seu mecanismo com o objetivo de se construir um gerador seguro do ponto de vista criptográfico. O PRNG é fundamentado em quatro blocos principais [60]:

- O mecanismo de geração de números pseudo-aleatórios, que utiliza o 3DES.
- Um acumulador de entropia, ou seja, um mecanismo que coleta amostras a serem utilizadas na produção de novas chaves criptográficas. A fonte destas amostras é determinada previamente e é uma característica particular da implementação.
- Um mecanismo para produção de novas chaves criptográficas.
- Um mecanismo de controle da produção de novas chaves criptográficas, responsável por determinar o momento em que ela deve ocorrer.

O bloco de geração de números pseudo-aleatórios do PRNG Yarrow-160 é o único de interesse para o mecanismo de Domínios Virtuais, já que a o processo de produção de novas chaves criptográficas, a princípio, não faz parte deste.

O algoritmo faz uso de um contador interno C_i e utiliza como bloco fundamental de funcionamento o 3DES, com três chaves de criptografia distintas[33] e está apresentado a seguir:

$$saída = E_K(C_i) \tag{12}$$

[33] O 3DES pode utilizar duas ou três chaves, o que acarreta chaves criptográficas equivalentes de 112bits e 168bits, consecutivamente [18].

$$C_{i+1} = (C_i + 1) \bmod 2^n \quad (13)$$

Onde *n* corresponde ao tamanho do contador e *K* ao conjunto de chaves criptográficas utilizadas pelo 3DES.

No entanto, não é possível adaptar o algoritmo para ser utilizado no mecanismo de Domínios Virtuais, pois faz uso de um contador incremental de *n* bits, cujo estado não deve ser revelado, e que é alterado a cada valor gerado. Este fato impossibilita o uso do mesmo em um grupo de usuários, já que o sincronismo dos valores gerados não pode ser obtido, pois não é possível contar com a atualização do contador em todos os dispositivos da rede móvel.

O processo de produção de novas chaves criptográficas não pode ser diretamente utilizado no mecanismo de Domínios Virtuais proposto, pois os valores de entropia coletados em cada um dos diversos dispositivos da rede móvel não são, certamente, os mesmos, o que acarretaria na pulverização do domínio virtual.

3.3.1.6 Considerações sobre os PRNG

Uma vez efetuado o estudo sobre a capacidade de se adequar os PRNG estudados ao mecanismo de Domínios Virtuais, pode-se concluir que apenas três, do total de cinco analisados, podem ser utilizados, sendo que apenas um deles sem que seja executada nenhuma modificação. A Tabela 3.1 apresenta os PRNG estudados e sua adequação ao mecanismo proposto.

Tabela 3.1: PRNG e sua adequação ao mecanismo de Domínios Virtuais.

	Adequado sem modificações	Adequado com modificações	Inadequado
ANSI X9.31		X	
DSA		X	
RSAREF			X
Blum Blum Shub	X		
Yarrow-160			X

O Blum Blum Shub é o único gerador que pode ser utilizado sem que exista a necessidade de uma intervenção em seu mecanismo de funcionamento,

pois seu algoritmo permite a utilização de um valor público e arbitrário, adequado para o parâmetro de tempo utilizado pelo sistema. No entanto, o Blum Blum Shub é, como descrito no item 3.3.1.4, computacionalmente intensivo, o que não é adequado para dispositivos móveis, já que o consumo de energia deve ser baixo, pois seu suprimento é, muitas vezes, extremamente limitado.

Tanto o ANSI X9.31 quanto o DSA necessitam de alterações em seus algoritmos, de forma a adequá-los aos requisitos do mecanismo de Domínios Virtuais. No entanto, o ANSI X9.31 pode produzir somente 64bits a cada iteração, enquanto que o algoritmo do DSA utiliza a função de *hash* SHA-1, podendo oferecer valores pseudo-randômicos de 160bits a cada iteração executada.

Sendo assim, a variante do PRNG DSA foi escolhida como gerador de números pseudo-aleatórios a ser implementado como parte componente do mecanismo de Domínios Virtuais, por poder oferecer um valor de comprimento suficiente para que possa ser dividido e utilizado em todas as trocas de mensagens efetuadas pelo mecanismo em apenas uma única iteração.

Os testes realizados sobre amostras produzidas pela variante do PRNG DSA utilizada no mecanismo de Domínios Virtuais são apresentados no item 5.2.

3.3.2 As Sementes

O gerador de números pseudo-aleatórios pode utilizar mais de uma semente, de modo a gerar seqüências diferentes, permitindo que um mesmo dispositivo pertença a vários domínios virtuais simultaneamente. Sendo assim, um dispositivo poderia utilizar uma semente para se comunicar dentro do domínio virtual correspondente ao ambiente doméstico de um usuário e outra para o domínio virtual do ambiente de trabalho do mesmo usuário.

As sementes nada mais são que um segredo compartilhado entre os dispositivos pertencentes a um domínio virtual e devem, portanto, estar protegidas, não podendo ser lidas ou acessadas por nenhum outro mecanismo que não o de Domínios Virtuais.

3.3.3 As Relações Temporais e o Relógio do Sistema

O controle sobre as relações temporais deve ser efetuado para evitar ataques de retransmissão de informações, ou seja, ataques *replay*. Este controle é feito através da operação conjunta do relógio do sistema e o bloco de verificação de relações temporais.

O controle de relações temporais define as chamadas janelas de tempo no qual o mecanismo irá aceitar mensagens de desafio. Uma janela de tempo pode ser definida como um intervalo temporal, no qual o valor atual de tempo do dispositivo, ou tempo local, corresponde ao centro deste intervalo, como observado na Figura 3.10.

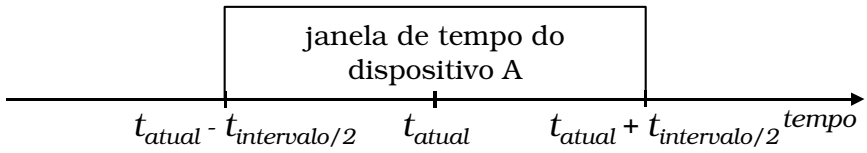

Figura 3.10: Janela de tempo do dispositivo A.

O recurso de janelas de tempo é utilizado devido à impossibilidade de se ter relógios naturalmente sincronizados nos diversos dispositivos que compõem a rede. Assim sendo, torna-se necessário determinar um período de tempo no qual as informações recebidas de outros dispositivos permanecem sendo consideradas válidas, evitando, assim, que mensagens antigas possam ser reutilizadas. Deste modo, mensagens recebidas que não possuem uma associação de tempo dentro da janela de transmissão são ignoradas pelo dispositivo receptor.

O tamanho, ou intervalo, desta janela de tempo está diretamente ligado à sincronização dos relógios dos dispositivos que compõem a rede móvel, ou seja, quanto maior a sincronização do relógio destes dispositivos, menor poderá ser esta janela de tempo.

Idealmente, poder-se-ia supor que todos os dispositivos da rede móvel ad hoc possuem um serviço NTP[34], o que permitiria a sincronização do relógio

[34] *Network Time Protocol* – padrão IETF para sincronização de relógios de equipamentos de rede a partir de uma referência de tempo [67].

dos diversos dispositivos que compõem a rede com o padrão internacional UTC[35]. No entanto, mesmo que todos os dispositivos móveis possuíssem um cliente NTP instalado, seria necessário que estes conseguissem, ao menos de tempos em tempos, se comunicar com um servidor NTP, o que não pode ser garantido em uma rede móvel ad hoc. Neste caso, a janela de tempo poderia ser a menor possível, ou seja, a soma da incerteza do NTP com o tempo de transmissão (tempo de processamento somado ao tempo de propagação) de um pacote para um dispositivo à máxima distância, subtraído do valor de tempo corrente do dispositivo, como apresentado na Figura 3.11.

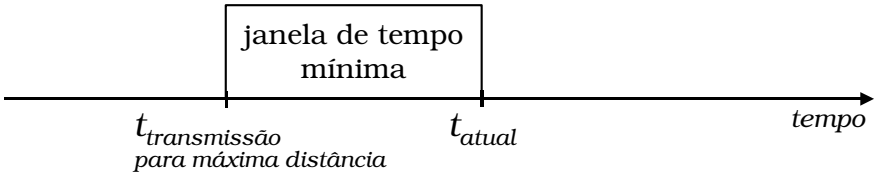

Figura 3.11: Intervalo mínimo de uma janela de tempo.

O tamanho da janela de tempo é também determinante do nível de segurança oferecido pelo mecanismo de Domínios Virtuais[36], pois define o intervalo de tempo no qual a rede móvel está susceptível a ataques de repetição, já que, um valor gerado em um determinado instante de tempo *t* em um dispositivo *x* deve ser igual ao produzido por um dispositivo *y* da mesma rede, e que possui, portanto, a mesma semente. Assim sendo, um atacante poderia capturar uma mensagem de desafio e tentar utilizá-la ainda dentro da janela de tempo válida, o que, entretanto, ainda não poderia prover ao atacante acesso ao dispositivo atacado, pois este ainda deveria ser capaz de enviar uma réplica e ainda responder a um desafio com um valor de tempo associado desconhecido a priori, a ser determinado pelo

---

[35] *Coordinated Universal Time* – escala de tempo padrão baseada no movimento de rotação da Terra e no calendário Gregoriano, é definida a partir de diversos relógios atômicos mantidos em diversos laboratórios de pesquisa espalhados em vários países.

[36] É possível notar que o menor período teoricamente aceitável de duração para uma janela de tempo é igual a duas vezes o tempo de transmissão de uma mensagem, ou seja, duas vezes o tempo de propagação da mensagem, somado a duas vezes o tempo de processamento da mesma, sendo que a janela de tempo deve ser posicionada somente à frente do valor de tempo atual.

dispositivo atacado. De modo a minimizar a ocorrência deste tipo de eventos dentro da rede móvel, o bloco de verificação de relação de tempo mantém um registro dos valores temporais já utilizados, levando em conta apenas os valores que pertencem e posteriores à sua janela temporal. Além disto, também é efetuado um controle sobre os endereços de origem de respostas e réplicas a desafios (ver item 3.3.4).

3.3.4 Verificação da Origem

O controle sobre a origem das respostas e das réplicas é feito com o intuito de evitar que mensagens consecutivas provenientes de uma mesma origem sejam consideradas dados válidos pelo mecanismo de Domínios Virtuais, já que este tipo de ocorrência indica, invariavelmente, uma tentativa de ataque. Sendo assim, mesmo que um dispositivo atacante envie a resposta correta para um determinado desafio, ou a réplica correta para uma determinada resposta após uma série de tentativas erradas, esta mensagem será desconsiderada pela segunda parte, através do monitoramento do endereço de origem das mensagens recebidas.

Além da verificação do endereço origem, também é verificada a existência de mensagens resposta duplicadas provenientes de endereços origem diferentes, ou seja, mensagens originadas de diferentes dispositivos, mas com os mesmos valores nos campos *tempo t2*, *valor gerado* x_i e *valor gerado* x_j. Essa ocorrência indica, fortemente, uma tentativa de ataque de repetição, já que é altamente improvável que dois diferentes dispositivos respondam a uma mesma mensagem *desafio* ao mesmo instante.

Sendo assim, o mecanismo deve aceitar apenas a primeira mensagem recebida, ignorando as demais que possuam os campos *tempo t2*, *valor gerado* x_i e *valor gerado* x_j repetidos, já para a ocorrência deste tipo de ataque é necessário que o atacante receba a mensagem original, substitua o campo *endereço origem*, e reenvie a mensagem, o que causaria um atraso mínimo igual ao tempo de processamento desta mensagem.

A freqüência de ocorrências deste tipo de tentativa de ataque depende do tamanho da janela de tempo utilizada pelo mecanismo, já que com o

estreitamento desta, este tipo de ocorrência torna-se, potencialmente, menos recorrente.

3.3.5 O Gerente e a Tabela de Mensagens Ativas

O gerente é responsável pela comunicação com a aplicação, recebendo da mesma um pedido de comunicação que especifica a rede a ser procurada, ou seja, uma rede doméstica ou uma rede empresarial, por exemplo, pois esta informação é determinante na escolha da semente a ser utilizada no PRNG. O valor pseudo-randômico, uma vez gerado, é então recebido pelo gerente e, então, repassado para o bloco de transmissão de dados.

O gerente também é responsável pela verificação dos campos de valores gerados de quaisquer mensagens recebidas que tenham passado com sucesso pela verificação de relações temporais e, no caso de mensagens do tipo *resposta* ou *réplica*, também pela verificação do endereço origem. Esta verificação é efetuada através da passagem dos valores temporais t_1 e, eventualmente, t_2 da mensagem recebida, assim como todas as sementes do mecanismo para o PRNG, que produz os números pseudo-aleatórios relacionados e repassá-os para o gerente, que é, então, capaz de confirmar a validade da mensagem recebida.

A última atribuição do gerente é a manutenção da tabelas de mensagens ativas, que contém uma relação das mensagens transmitidas do tipo *desafio* ou *resposta* com o seu tempo de vida, que determina a validade de uma mensagem, e valores pseudo-randômicos restantes esperados, previamente calculados pelo PRNG ao comando do gerente. O tempo de vida de uma mensagem enviada é, no mínimo, igual a duas vezes o tempo de transmissão de uma mensagem para um dispositivo à máxima distância permitida, como apresentado na Figura 3.12, sendo que o tempo de vida de uma mensagem do tipo *desafio* é maior, já que a produção de uma mensagem do tipo *resposta* exige que o PRNG produza dois valores pseudo-aleatórios distintos, enquanto que a produção de uma mensagem do tipo *réplica* não exige a produção de nenhum outro valor que não possa ter sido previamente gerado.

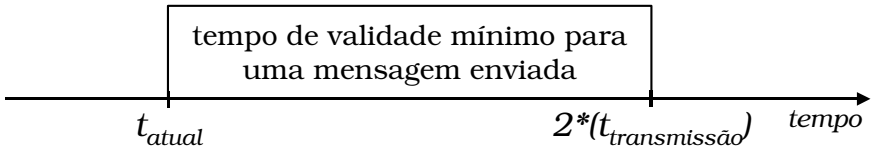

Figura 3.12: Tempo de validade mínimo para uma mensagem enviada.

Mensagens recebidas do tipo *resposta* ou *réplica*, que passaram com sucesso pela verificação de relações temporais e de endereço origem, são tratadas pelo gerente somente se existir uma correlação entre estas e as mensagens presentes na tabela de mensagens ativas.

O gerente também verifica a existência da duplicação de mensagens do tipo *desafio*, o que indica fortemente uma tentativa de ataque do tipo *man-in the-middle*[37]. Mensagens duplicadas são retiradas da tabela de mensagens ativas e, portanto, mensagens do tipo *resposta* associadas a estas são ignoradas.

Este procedimento é capaz de evitar uma parte significativa de ataques *man-in-the-middle*, já que as principais tecnologias de redes móveis locais, como o IEEE 802.11 [6] e o Bluetooth [8] utilizam antenas omnidirecionais, ou seja, os sinais eletromagnéticos são transmitidos em todas as direções. Deste modo, para que um ataque *man-in-the-middle* tenha sucesso, o atacante teria que deixar a área de alcance do dispositivo que transmitiu a mensagem original, retransmitir a mensagem capturada, esperar a resposta da mesma, e então retornar à área de alcance do primeiro dispositivo em um tempo inferior ao de validade da mensagem enviada.

3.3.6 Transmissor/Receptor

Os blocos transmissão e recepção são responsáveis, respectivamente, pela saída e pela entrada, respectivamente, de informação no mecanismo de Domínios Virtuais relativas ao meio externo, através do envio e recepção de dados para as interfaces de comunicação presentes no dispositivo. Tais interfaces podem ser de qualquer tecnologia de comunicação para redes

[37] O ataque *man-in-the-middle* se caracteriza pela captura de uma mensagem pelo dispositivo atacante e posterior retransmissão da mesma, com o dispositivo atacante assumindo o papel do dispositivo original.

móveis, desde que sejam capazes de enviar as mensagens definidas na arquitetura do mecanismo de Domínios Virtuais, como mensagens do tipo *broadcast* e também mensagens com endereço origem vazio.

3.4 O Mecanismo em seu Contexto

O mecanismo de Domínios Virtuais pode ser implementado junto a outros mecanismos ou modelos de segurança, agregando funcionalidades e, portanto, contribuindo com o incremento do nível de segurança oferecido ao usuário.

Os maiores benefícios da união do mecanismo de Domínios Virtuais com outros modelos e mecanismos de segurança são alcançados através do uso de autenticação utilizando algoritmos de chave pública. O fortalecimento do mecanismo também é possível através da agregação de políticas e processos de distribuição e renovação de sementes.

3.4.1 Chaves Públicas: Autenticação e Anonimato

Um processo de autenticação utilizando criptografia de chaves públicas [68], ou assimétrica, sendo executado logo após o funcionamento do mecanismo de Domínios Virtuais além de simplesmente oferecer um segundo obstáculo a eventuais atacantes é também capaz de eliminar completamente a ocorrência de ataques do tipo *man-in-the-middle*[38], independente do tempo de validade dado às mensagens transmitidas, já que autentica univocamente as partes envolvidas na comunicação.

3.4.1.1 A criação do anonimato completo

O mecanismo de Domínios Virtuais, sozinho, é capaz de oferecer um anonimato superior ao oferecido pela tecnologia Bluetooth[39] (ver item 2.3.4) já que, para uma terceira parte, só é possível identificar um dispositivo a depois que este inicia uma busca por outros pertencentes ao seu domínio

---

[38] Assumindo que as entidades certificadoras que produziram os certificados em questão sejam reconhecidas por ambas as partes envolvidas no processo de autenticação.

[39] Na tecnologia Bluetooth, o anonimato termina a partir do instante em que o dispositivo inicia uma requisição de serviços [42].

virtual, enviando uma mensagem do tipo *desafio*, obtendo então uma mensagem *resposta*, necessariamente correta, ou seja, deve ser enviada por outro dispositivo pertencente ao mesmo domínio virtual, para então, ter sua identidade revelada no envio de uma mensagem do tipo *réplica*.

A utilização de uma variante do mecanismo de Domínios Virtuais seguido de um processo de autenticação utilizando chaves públicas é capaz de oferecer o anonimato completo sem a utilização de elementos centrais. Isto é possível através da criação e utilização de pseudônimos, que utilizados nos campos de endereço origem e destino das mensagens trafegadas, associado à criptografia de certificados utilizando uma chave simétrica determinada durante a troca inicial de mensagens, impede uma terceira parte, através de um ataque passivo, de identificar os dispositivos participantes da comunicação, e viabiliza o completo anonimato dos dispositivos.

A chave simétrica utilizada para a troca de certificados pode ser obtida através da geração de um novo número pseudo-randômico, utilizando a semente associada a um dos valores previamente gerados ou ainda uma parte não utilizada dos valores pseudo-aleatórios, como entrada do PRNG, por exemplo.

3.4.1.2 Considerações sobre o consumo de energia

Energia ainda é um recurso extremamente limitado em dispositivos portáteis, o que limita a utilização de recursos que demandam uma grande quantidade de processamento e, portanto, de energia, como a criptografia de chaves públicas.

No entanto, a utilização de criptografia de chaves públicas em conjunto com o mecanismo de Domínios Virtuais abre a possibilidade para a diminuição da utilização de criptografia assimétrica, pois esta passa a ser restrita aos dispositivos que passarem pelo mecanismo de Domínios Virtuais, diminuindo, portanto, o gasto desnecessário de energia em operações mal-sucedidas de criptografia de chaves públicas.

3.4.2 A Geração e Distribuição das Sementes

As sementes nada mais são que um segredo compartilhado entre os diversos dispositivos que compõem um domínio virtual, como apresentado no item 3.3.2. As políticas aplicadas na geração e na distribuição das sementes são cruciais para a garantia da segurança do mecanismo, já que a utilização de uma semente fraca ou o uso de um processo de distribuição de sementes falho podem acabar comprometendo a segurança oferecida pelo mecanismo.

A geração das sementes necessita de um ou mais dispositivos que determinem, ou seja, produzam uma semente a ser utilizada dentro do domínio virtual. Como não é possível garantir que todos os dispositivos do domínio virtual estejam presentes durante este processo, a geração da semente pode adotar uma solução centralizada, na qual um dispositivo gera a semente sem a participação de outros dispositivos, ou uma solução contributiva utilizando os dispositivos presentes [43] [44].

A distribuição das sementes entre os dispositivos pertencentes à rede deve ser feita a partir de um ou mais dispositivo considerados seguros. A abordagem utilizando apenas um dispositivo para a distribuição de um segredo compartilhado é utilizada pelo modelo de segurança *Resurrecting Duckling* [19] [21], no qual um dispositivo especial, o representante cibernético, é responsável pela distribuição das sementes[40] através de contacto físico com o dispositivo a ser incluído no domínio virtual. A replicação do dispositivo confiável em múltiplos dispositivos confiáveis a serem utilizados na distribuição de um segredo compartilhado é feita no Modelo de Segurança para Redes Ad Hoc [15] [16], apresentado no item 2.3.7.

3.4.3 Renovação Automática das Sementes

O processo de renovação de sementes utilizadas pelo PRNG é fundamental para a manutenção da segurança oferecida pelo mecanismo de Domínios

---

[40] No artigo, certificados são distribuídos deste modo, e não sementes para um PRNG.

Virtuais. O processo de renovação manual das sementes não é adequado, pois exige a intervenção do usuário, que, como colocado no item 2.2, não são especialistas em configuração de dispositivos, além de consistir em uma tarefa inevitavelmente repetitiva, além de obrigar o usuário a se familiarizar com interfaces de diversos dispositivos.

A renovação automática é, portanto, fundamental para o funcionamento adequado do mecanismo de Domínios Virtuais, e pode ser executada através da utilização de uma rede orientada a serviços [15] [16], no qual um dos Serviços de Registro responsáveis pela emissão de certificados dentro da rede gera uma nova semente e, após assumir o papel de serviço de diretório[41] da rede orientada a serviços, inicia o processo de substituição das sementes antigas pelas novas, durante o qual uma verificação dos dispositivos em questão é feita, de modo que a troca só é executada caso não exista nenhum impedimento para tal, como a presença em uma eventual lista de revogação de certificados, por exemplo.

Outro solução para a renovação de sementes pode ser retirada do Modelo de Segurança *Resurrecting Duckling* [19] [21], no qual um dispositivo confiável aos demais participantes da rede, como o representante cibernético, poderia ser utilizado. Esta solução, no entanto, não permite uma renovação automática das sementes, exigindo que o usuário efetue a troca manualmente em cada um de seus dispositivos, o que, claramente, não é uma solução adequada.

---

[41] O serviço de diretórios é utilizado para anunciar aos diversos dispositivos que compõem a rede quais são os serviços disponíveis na mesma. A utilização de um serviço de diretórios não é, entretanto, um consenso para tecnologias de redes orientadas a serviços, sendo uma entidade obrigatória no Jini da SUN Microsystems [69]. Entretanto, o mecanismo proposto continua válido mesmo para outras tecnologias de redes orientadas a serviços que não dependem de um serviço de diretórios, como o UPnP [70], pois todo dispositivo possui uma relação completa dos serviços disponíveis na rede.

Capítulo 4

Implementação

"In theory, there is no difference between theory and practice, but in practice, there is."

Jan van de Snepscheut

A descrição do processo de implementação de um protótipo do mecanismo de Domínios Virtuais, proposto nesta dissertação e seus aspectos relacionados, como a máquina de estados e o ambiente de desenvolvimento utilizado, por exemplo, correspondem ao objetivo deste capítulo.

Deste modo, este capítulo apresenta inicialmente os aspectos gerais da implementação, de modo a contextualizar o seu desenvolvimento dentro do escopo ao qual pertence a aplicação desenvolvida. A máquina de estados do mecanismo é então apresentada, sendo seguida pelo seu diagrama de classes.

4.1 Aspectos Gerais da Implementação

Aspectos gerais da implementação, como o escopo da aplicação, o ambiente de desenvolvimento utilizado e as interfaces do mecanismo são descritos neste item, que tem por objetivo contextualizar a implementação do mecanismo de Domínios Virtuais dentro dos objetivos desta dissertação e de seu propósito dentro de uma ampla arquitetura de segurança voltada às redes móveis ad hoc.

4.1.1 O Escopo da Implementação

A implementação do mecanismo de Domínios Virtuais faz parte, como já explicitado no item 1.3, de uma ampla arquitetura, pertencente a um modelo de segurança para redes móveis ad hoc, correspondendo o mecanismo proposto a um módulo desta arquitetura.

Assim sendo, muitos dos aspectos fundamentais do desenvolvimento desta aplicação estão intimamente correlacionados a decisões tomadas considerando o mecanismo como parte de um todo, no qual o todo corresponde à arquitetura em sua plenitude.

Deste modo, é possível definir claramente o papel do mecanismo de Domínios Virtuais, que é fornecer uma proteção inicial à rede móvel como um todo, impedindo atacantes externos de obterem informações sobre os dispositivos pertencentes a um domínio de uma rede móvel.

4.1.2 Ambiente e Linguagem de Programação

A linguagem de programação escolhida para o desenvolvimento da aplicação foi Java. Esta escolha foi influenciada definitivamente pelas vantagens oferecidas pelo ambiente Java.

Entre as vantagens de maior relevância para a escolha do Java como plataforma de desenvolvimento, está a geração de código independente de plataforma, ou seja, da arquitetura do *hardware* e sistema operacional do dispositivo, já que se trata de uma linguagem interpretada por uma máquina virtual, a chamada máquina virtual Java (JVM[42]). Estas características são especialmente desejáveis para equipamentos de eletrônica de consumo, que correspondem a um grupo de dispositivos que poderiam ter o benefício da utilização do mecanismo de Domínios Virtuais, permitindo a identificação pelo dispositivo de pares confiáveis dentro de um ambiente que utilize redes móveis ad hoc. Além disso, a existência de uma arquitetura de segurança nativa, que permite a execução de aplicações de forma segura também foi outra vantagem relevante da linguagem Java que a levou a ser escolhida para o desenvolvimento do projeto [71].

Esta escolha, no entanto, inicialmente limita a aplicabilidade do protótipo, já que o Java trata toda troca de dados que utilize uma rede de computadores como uma propriedade da camada de transporte, ou seja, a

---

[42] *Java Virtual Machine*

comunicação é executada apenas através do uso dos protocolos TCP[43] e UDP[44]. Deste modo, o mecanismo de Domínios Virtuais pode ser utilizado somente se houver o conhecimento prévio de informações sobre os demais dispositivos da rede, como um endereço lógico, ou o nome de um dispositivo, caso exista algum mecanismo de tradução de nomes presente. Esta restrição da aplicabilidade do protótipo, no entanto, não interfere na validação conceitual do mecanismo, que é, na verdade, o propósito fundamental de sua construção.

É importante ressaltar que esta limitação pode ser contornada, em alguns casos, através da utilização de módulos codificados em outras linguagens e que tenham acesso à interface de comunicação, como o pacote WinPcap, desenvolvido em linguagem C para plataformas Win32 [72]. Estes módulos podem ser encapsulados em uma classe Java e então utilizados.

Uma vez definido o Java como plataforma, o ambiente de desenvolvimento escolhido, dentre os muitos existentes para esta linguagem, foi o SUN ONE Studio4, também denominado de Forte 4.0 CE. A versão da plataforma Java utilizada foi a J2SDK SE v.1.4.0.

4.1.3 Interface do Mecanismo

O mecanismo de Domínios Virtuais, por tratar-se de um módulo presente em uma ampla arquitetura de segurança para redes móveis ad hoc, não possui uma interface homem-máquina elaborada, que, além de desnecessária para um mecanismo com este propósito, somente causaria o aumento do tamanho do código produzido, que é, muitas vezes, um fator limitante em dispositivos portáteis, já que estes nem sempre possuem recursos abundantes.

Deste modo, a interface do mecanismo corresponde, na verdade, ao acesso a seus métodos públicos, de forma a permitir que o mesmo seja agregado a outros aplicativos ou ainda possa ser executado de modo independente,

---

[43] *Transmission Control Protocol* - protocolo de comunicação de camada de transporte orientado à conexão.
[44] *User Datagram Protocol* - protocolo de comunicação de camada de transporte não orientado à conexão.

através de linhas de comando. Para efeito de demonstração, uma interface simples foi adicionada ao mecanismo de forma a facilitar e permitir a sua apresentação como uma aplicação isolada

4.1.4 Redes Wireless

O mecanismo de Domínios Virtuais foi aplicado, para ser posteriormente testado, em uma rede móvel formada por dois computadores portáteis utilizando tecnologia de comunicação sem fio IEEE 802.11b em modo ad hoc, de modo que ambos os dispositivos componham uma IBSS[45] [6] [7].

4.2 Máquina de estados

A máquina de estados do mecanismo de Domínios Virtuais é definida a partir da seqüência de mensagens trocadas pelo protocolo, como apresentado previamente no item 3.2.2.3 e também na Figura 3.1. Podem ser identificados oito estados distintos, que são nomeados na Tabela 4.1.

Tabela 4.1: Os estados da máquina.

Estado	Nome do Estado
0	Esperando envio ou recebimento de mensagens *desafio*.
1	Esperando mensagem *resposta*.
2	Verificando mensagem *desafio*.
3	Esperando mensagem *réplica*.
4	Verificando mensagem *resposta*.
5	Esperando mensagem OK.
6	Verificando mensagem *réplica*.
7	Conectado

A máquina de estados é apresentada na Figura 4.1, na qual os arcos representam as transições ocorridas devido à chegada, transmissão ou avaliação de uma mensagem.

Os arcos contínuos correspondem às transições percorridas pelo mecanismo de Domínios Virtuais quando este assume o papel de cliente, ou

[45] *Independent Basic Service Set*

seja, envia uma mensagem do tipo *desafio*, na tentativa de verificar se o dispositivo contatado pertencente a um domínio virtual conhecido. Os arcos não-contínuos, por sua vez, correspondem às transições percorridas pelo mecanismo quando este assume o papel de servidor, ou seja, recebe uma mensagem do tipo *desafio*.

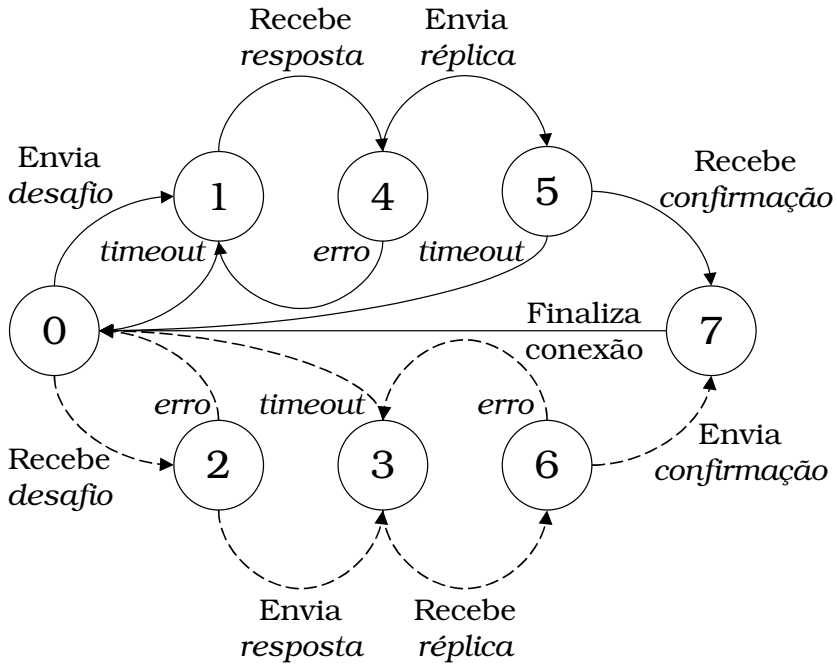

Figura 4.1: Máquina de estados do mecanismo de Domínios Virtuais.

A máquina de estados pode, como apresentado, assumir dois caminhos distintos, pois o mecanismo possui características tanto de cliente como de servidor, correspondendo, portanto, a uma aplicação *peer-to-peer*[46].

O retorno de um estado para outro anterior a ele pode ser acarretado pela ocorrência de um *timeout*, de um *erro* ou ainda através da finalização da conexão após um procedimento bem-sucedido, ou seja, somente após o reconhecimento de ambas as partes como pertencentes a um mesmo domínio virtual.

A ocorrência de um evento de *timeout*, previsto na máquina de estados do mecanismo, acontece sempre que uma mensagem é descartada por exceder

[46] O modelo *peer-to-peer* existe em oposição ao modelo cliente/servidor. O modelo *peer-to-peer* pode ser, resumidamente, definido como um modelo no qual os pares são, simultaneamente, clientes e também provedores de serviço.

o limite de sua validade no domínio do tempo, ou seja, localizar-se além da janela temporal definida à mesma (como visto no item 3.3.3). A ocorrência de um *timeout* sempre acarreta o retorno da máquina para o seu estado inicial (0).

Um evento *erro* ocorre toda vez que uma mensagem recebida pelo mecanismo não possui conteúdo válido, ou seja, os dados contidos na mesma não correspondem a valores esperados. Um *erro* pode ainda ser causado por mensagens recebidas que contenham informações utilizadas previamente em outras recebidas anteriormente, o que indicaria com alta probabilidade a tentativa de um ataque do tipo *man-in-the-middle* (como visto no item 3.3.5).

A ocorrência de um evento *erro* sempre acarreta o retorno de um estado para o estado imediatamente anterior a este, de acordo com o diagrama apresentado na Figura 4.1. Deste modo, a ocorrência de um evento *erro* resultaria no retorno do estado dois (2) para o estado inicial zero (0), ou ainda do estado quatro (4) para o estado um (1), por exemplo.

4.3 Campos das Mensagens

Os campos que compõem cada uma das mensagens trocadas pelo mecanismo, descritos anteriormente no item 3.2.2.5, são detalhados neste item. Este detalhamento é feito através da especificação da origem das informações contidas nestes campos, de eventuais códigos utilizados na sua identificação e de características da construção dos mesmos, como o comprimento do campo em bits.

4.3.1 Tipos de Mensagens

Como especificado anteriormente no item 3.2.2.4, todas mensagens trocadas entre os dispositivos durante o processo de reconhecimento possuem um campo *tipo* associado, que indica qual o gênero da mensagem, ou seja, se a mensagem em questão é do tipo *desafio*, *resposta* ou *réplica*. Além destes, um quarto tipo de mensagem, apenas para a confirmação de que o procedimento ocorreu de forma correta, também é previsto. Esta

mensagem final é denominada de mensagem do tipo *confirmação* ou, simplesmente, mensagem *OK*. Os tipos das mensagens são determinados por um código valor hexadecimal, conforme apresentado na Tabela 4.2.

Este campo possui 8bits de comprimento.

Tabela 4.2: Tipos de mensagens e seus códigos.

Tipo da mensagem	Código da mensagem
Mensagem *desafio*	0x01
Mensagem *resposta*	0x02
Mensagem *réplica*	0x03
Mensagem *confirmação*	0x04

Mensagens recebidas pelo mecanismo que possuam um campo *tipo* não previsto na Tabela 4.2 são descartadas pelo mecanismo.

4.3.2 Número de Seqüência

Especificado no item 3.2.2.4, todas mensagens trocadas entre os dispositivos possuem um campo *seqüência* que associa uma mensagem a um valor numérico, sendo que este indica o valor esperado para a próxima mensagem esperada de uma seqüência, ou seja, se uma mensagem do tipo *desafio* for enviada por um dispositivo com um número de seqüência *n*, o campo *seqüência* da mensagem do tipo *resposta* associada deve conter o valor *n+1*. Uma troca de mensagens entre dois dispositivos é apresentada na Figura 4.2, apresentando o comportamento do campo *seqüência* frente ao campo *tipo*.

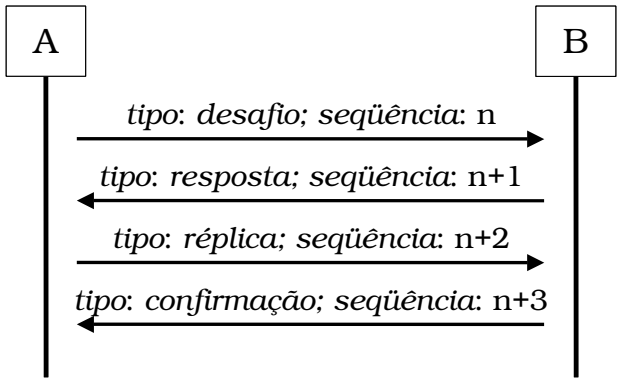

Figura 4.2: Troca de mensagens e os campos *tipo* e *seqüência*.

O valor de seqüência é de extrema importância para o mecanismo de Domínios Virtuais, pois as mensagens são armazenadas de acordo com o número de seqüência atribuído a ela. Assim, quando um dispositivo recebe uma mensagem do tipo *resposta* com um número de seqüência *n+1*, o mecanismo procura no índice de mensagens enviadas do tipo *desafio*, uma mensagem que apresente o número de seqüência *n*, antes de realizar qualquer verificação dos demais campos.

O campo *seqüência* possui 32bits de comprimento. Seu valor é gerado aleatoriamente utilizando uma classe nativa da plataforma de desenvolvimento, que implementa um PRNG de 32bits. A utilização de um PRNG para a determinação dos valores de seqüência tem, como objetivo, evitar que exista colisão destes dentro da área de alcance do dispositivo e assim reduzir a quantidade de processamento desnecessário de mensagens dos tipos *resposta* e *réplica*, já que toda mensagem é identificada pelo número de seqüência.

Um universo de 2^{32} valores possíveis praticamente elimina a ocorrência de colisões de campos *seqüência*, já que as tecnologias de redes móveis existentes apresentam áreas de alcance limitadas, de algumas dezenas de metros nos melhores casos.

4.3.3 Tempos

Os campos relativos aos valores de tempo utilizados para a geração dos números pseudo-aleatórios utilizados pelo mecanismo de Domínios Virtuais possuem, cada um, 64bits de comprimento.

As informações de tempo são dadas em milésimos de segundo e calculadas a partir da diferença entre o valor atual obtido do relógio do sistema e a meia-noite de primeiro de janeiro de 1970, no padrão de tempo da UTC.

4.3.4 Valores Gerados

Os campos da mensagem, que contêm os valores gerados pelo PRNG, possuem 40bits de comprimento cada um. Assim sendo, utilizando o PRNG DSA, como definido no item 3.3.1.6, são utilizados, no máximo, 120bits de cada valor de 160bits gerado, já que são três as mensagens utilizadas pelo

mecanismo que transportam valores produzidos pelo gerador pseudo-randômico.

Deste modo, uma mensagem do tipo *desafio* contém os primeiros 40bits do primeiro valor gerado, enquanto que a mensagem do tipo *resposta* associada a ela transporta os 40bits seguintes deste valor e ainda os 40bits iniciais de um segundo valor produzido. Por último, a mensagem do tipo *réplica* contém os terceira parte do primeiro valor e a segunda parte do segundo valor, ambas de comprimento 40bits.

A relação entre o índice dos bits transmitidos de cada um dos valores gerados pelo PRNG e os diversos tipos de mensagem é apresentada na Tabela 4.3.

Tabela 4.3: Tipos de mensagens e bits transmitidos de cada valor gerado.

Tipo da mensagem	Índice dos bits transmitidos do 1º valor gerado	Índice dos bits transmitidos do 2º valor gerado
desafio	01 – 40	-
resposta	41 – 80	01 – 40
réplica	81 – 120	41 – 80
confirmação	-	-

É importante ressaltar que no mínimo 40bits de cada número gerado não são utilizados. Este fato impede que um ataque passivo possa obter uma informação completa sobre um valor pseudo-aleatório produzido, reduzindo a probabilidade de sucesso de um eventual processo de criptoanálise, que faça uso de um ataque de força bruta, à procura da semente utilizada pelo mecanismo.

4.4 Casos de Uso

A definição do caso de uso tem como objetivo descrever como uma entidade usuário, ou ator[47], é capaz de interagir com a implementação do mecanismo

[47] Um ator é uma entidade externa ao sistema que participa do caso de uso, estimulando o mesmo com eventos de entrada.

de Domínios Virtuais. Apenas dois casos de uso são previstos para o mecanismo:

- Identificar dispositivos que pertençam a um domínio virtual conhecido.
- Alterar parâmetros do mecanismo.

Para identificar outros dispositivos que pertençam a um domínio virtual conhecido, o ator inicia o processo definindo o nome do domínio virtual previamente cadastrado a ser procurado. O nome do domínio também pode ser escolhido pelo ator a partir de uma lista, que pode ser apresentada, caso necessário.

A alteração dos parâmetros do mecanismo de Domínios Virtuais inclui a alteração da porta de comunicação TCP definida, a inclusão de sementes, que podem ser geradas automaticamente ou definidas arbitrariamente pelo ator, e ainda a alteração de propriedades de segurança, como o tamanho da janela de tempo utilizada.

4.5 Especificação de Classes

A partir da especificação e da arquitetura do projeto, apresentadas no itens 3.2 e 3.3 respectivamente, foi possível definir as classes necessárias para a implementação do mecanismo de Domínios Virtuais.

Deste modo, quatro classes foram especificadas e, posteriormente, desenvolvidas. Estas classes são:

- Classe *SeedTable*, utilizada para o armazenamento das sementes.
- Classe *MessageIndex*, que armazena as mensagens transmitidas.
- Classe *PRNG*, responsável pela geração de valores aleatórios.
- Classe *AuthNetwork*, responsável pelas interfaces do mecanismo com o meio externo, pela construção das mensagens a serem transmitidas e pela lógica da avaliação das mensagens recebidas.

Além destas, foram adicionadas a esta implementação três outras classes para o cálculo da função de *hash* SHA-1 [73].

4.5.1 Diagrama de Classes

O diagrama de classes, ilustrando suas relações, é apresentado na Figura 4.3.

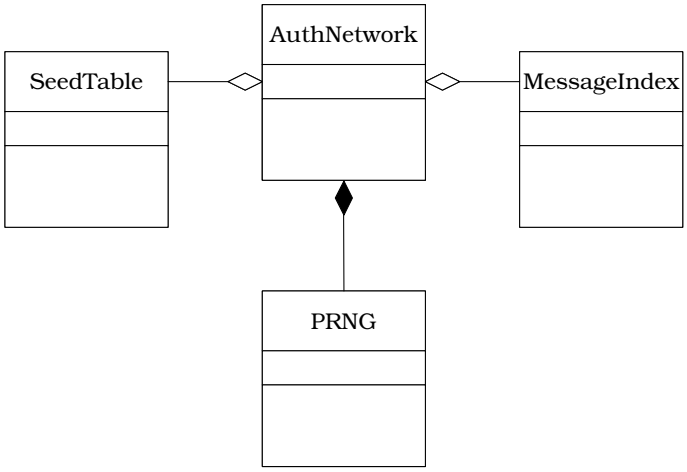

Figura 4.3: Diagrama de classes do mecanismo de Domínios Virtuais.

As classes adicionais responsáveis pelo cálculo da função de *hash* SHA-1 foram suprimidas do diagrama, já que se tratam de classes de uso público, e não foram desenvolvidas com o objetivo exclusivo de implementar o mecanismo proposto nesta dissertação. Do mesmo modo, outras classes utilizadas pertencentes à plataforma de desenvolvimento Java também não foram explicitadas no diagrama apresentado na Figura 4.3.

4.5.2 Descrição das Classes

A descrição detalhada das funcionalidades de cada uma das classes implementadas é apresentada no decorrer deste item, assim como as suas responsabilidades e eventuais limitações.

4.5.2.1 Classe AuthNetwork

Esta pode ser considerada a principal classe do mecanismo, já que as demais são instanciadas somente a partir da instanciação desta, como apresentado no diagrama da Figura 4.3.

A classe *AuthNetwork* agrega funções de quatro dos blocos da arquitetura do mecanismo, apresentada no item 3.3, incluindo as interfaces do

mecanismo, que são os dois blocos de comunicação, transmissor e receptor, e o gerente.

O quarto bloco da arquitetura implementado nesta classe é, naturalmente, a verificação das relações temporais das mensagens recebidas, já que este bloco localiza-se entre o receptor e o gerente.

A avaliação das mensagens recebidas é também executada por esta classe, que efetua inicialmente uma triagem das mesmas a partir de seu campo *tipo*. Além disso, a montagem das mensagens a serem enviadas também é de responsabilidade desta classe.

A recepção de mensagens foi implementada através da execução de uma *thread*, que aguarda conexões em uma porta TCP, enquanto que a transmissão é executada através do envio de dados à mesma porta TCP.

A verificação das relações temporais é efetuada a partir da comparação do valor de tempo incluído na mensagem recebida e um intervalo ao redor do valor de tempo corrente do sistema. Esta verificação é executada antes de qualquer outra verificação dos valores contidos na mensagem, já que está baseada em apenas quatro operações simples, sendo uma soma, uma subtração e duas comparações. Deste modo, pode-se evitar a execução desnecessária de outras funções aritmeticamente mais complexas, que exijam uma maior quantidade de processamento e conseqüente gasto de energia, como a geração desnecessária de números pseudo-randômicos.

4.5.2.2 Classe SeedTable

A classe *SeedTable* é responsável pelo armazenamento das sementes utilizadas pelo PRNG, sendo que cada uma produz uma seqüência de valores diferente e, portanto, identifica um único domínio virtual.

A tabela é indexada por nomes que identificam cada uma das sementes, sendo que não é possível a existência de nomes duplicados. O nome da semente tem valor apenas dentro do contexto local da aplicação, podendo a mesma semente possuir nomes diferentes em dois dispositivos distintos. De fato, o nome da semente serve apenas como um indicador para o usuário, de forma a facilitar a associação das sementes aos diferentes domínios

virtuais, como o “domínio do escritório”, ou o “domínio de casa”, por exemplo.

4.5.2.3 Classe MessageIndex

A classe *MessageIndex* armazena uma relação das mensagens envidas pelo dispositivo, separando-as em três tabelas distintas de acordo com o seu tipo, ou seja, uma tabela para mensagens enviadas do tipo *desafio*, outra para mensagens do tipo *resposta* e uma terceira para aquelas que possuam o tipo *réplica*.

Mensagens de confirmação transmitidas não são armazenadas em nenhum instante, pois nenhuma outra mensagem é associada a uma mensagem do tipo *confirmação*, ou seja, nenhuma outra mensagem pertencente ao mecanismo de Domínios Virtuais deve ser recebida após esta, uma vez que o processo dá-se por finalizado.

As tabelas são indexadas pelo número de seqüência das mensagens transmitidas, e são armazenadas imediatamente antes do processo de transmissão. Uma mensagem pode ser excluída da tabela de dois modos distintos:

- Pelo recebimento de uma mensagem esperada associada a esta. Por mensagem esperada entende-se que o tipo desta, assim como o seu número de seqüência e valores pseudo-randômicos gerados estejam todos corretos. É igualmente esperado que eventuais novos índices de tempo recebidos, no caso de mensagens do tipo *resposta*, estejam dentro dos parâmetros definidos pela janela de tempo aplicada.
- Pela ocorrência de um evento de *timeout*, ou seja, a extrapolação dos limites da janela de tempo definida acarreta a perda da validade das informações contidas em uma mensagem armazenada, causando a sua exclusão da tabela.

O número de seqüência é utilizado como índice das tabelas de mensagens transmitidas, pois este se trata de um valor seqüencial e está associado a uma mensagem recebida anteriormente, com exceção das mensagens do tipo *desafio*. Assim, por exemplo, quando uma mensagem do tipo *resposta* é

recebida por um dispositivo qualquer, este verificará a sua tabela de mensagens transmitidas do tipo *desafio* à procura de um índice que, quando somado a uma unidade, seja igual ao número de seqüência recebido.

4.5.2.4 Classe PRNG

A classe *PRNG* é responsável pela geração de valores pseudo-randômicos, tendo como parâmetros de entrada uma semente e o valor de tempo corrente, obtido do relógio do sistema, como descrito no item 4.3.3. Além disso, produz novas sementes, utilizando como parâmetros de entrada um nome definido pelo usuário e o tempo atual.

4.6 Diagramas de Interação

A representação de cada um dos casos de uso definidos no item 4.4 através da utilização de diagramas de interação permite que seja possível compreender melhor a relação entre as instâncias do diagrama de classes, definido no item 4.5.1, através da ilustração das mensagens trocadas entre eles.

4.6.1 Alteração de Parâmetros do Mecanismo

Alguns parâmetros presentes no mecanismo podem ter seus valores alterados, de modo a personalizar a aplicação de acordo com os requisitos definidos pelo usuário. O acréscimo e remoção de domínios virtuais do mecanismo também são operações apresentadas neste item.

4.6.1.1 Alteração da porta TCP

O usuário pode alterar a porta TCP utilizada para a comunicação do mecanismo de Domínios Virtuais. A aplicação retorna ao usuário uma mensagem comunicando a realização da alteração, indicando a nova porta de comunicação TCP escolhida.

O diagrama de seqüência representando este caso de uso está apresentado na Figura 4.4.

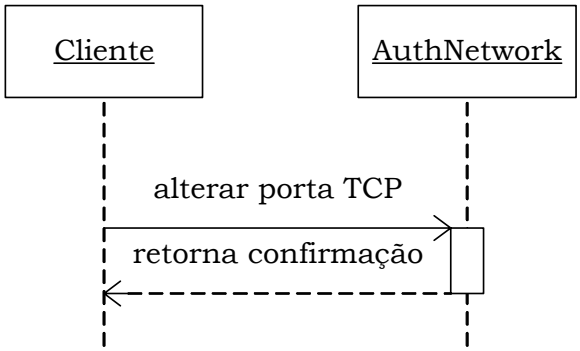

Figura 4.4: Alteração da porta TCP.

4.6.1.2 Redefinição do tamanho da janela de tempo

O usuário pode redefinir o tamanho da janela de tempo utilizada pelo mecanismo, de modo que este possa ser adequado aos seus requisitos individuais de segurança e também à capacidade dos dispositivos presentes no domínio virtual. A aplicação retorna ao usuário uma mensagem comunicando a realização da alteração, indicando o novo tamanho definido para a janela de tempo.

O diagrama de seqüência representando este caso de uso está apresentado na Figura 4.5.

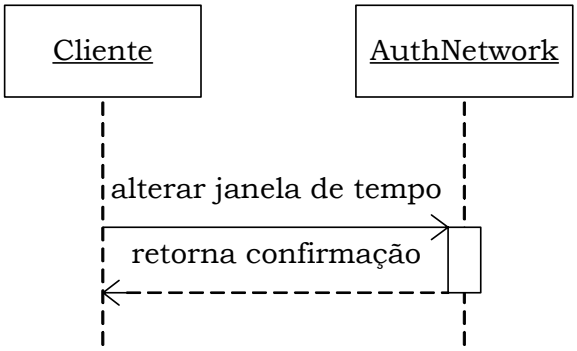

Figura 4.5: Redefinição da janela de tempo.

4.6.1.3 Inclusão de um domínio virtual

O usuário pode incluir um novo domínio virtual na sua relação de domínios virtuais conhecidos, através da definição de um nome para este domínio e da escolha de uma semente a ser utilizada no PRNG, que então será capaz de produzir as seqüências de valores pseudo-randômicos que irão definir o endereço deste domínio virtual em um determinado instante.

O objeto instanciado da classe *AuthNetwork* executa um método do objeto instanciado da classe *SeedTable* para o armazenamento da semente, utilizando o nome do domínio virtual e o valor da semente, definidos pelo usuário, como parâmetros deste método.

Ao final do processo, o objeto oriundo da classe *SeedTable* comunica ao objeto da classe *AuthNetwork* a realização da operação. A aplicação então retorna ao usuário uma mensagem comunicando a realização da inclusão.

O diagrama de seqüência representando este caso de uso está apresentado na Figura 4.6.

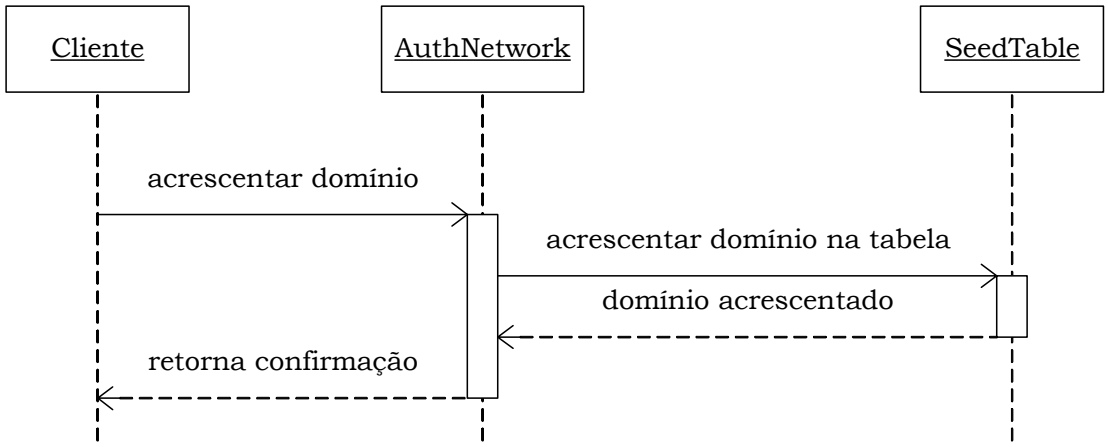

Figura 4.6: Inclusão de um domínio virtual.

4.6.1.4 Exclusão de um domínio virtual

O usuário pode excluir um determinado domínio virtual da sua relação de domínios virtuais cadastrados, através da escolha do nome do domínio virtual a ser excluído.

O objeto instanciado da classe *AuthNetwork* executa um método do objeto instanciado da classe *SeedTable* para a exclusão da semente do domínio virtual, utilizando como parâmetro deste método, o nome do domínio virtual definido pelo usuário.

Ao final do processo de exclusão, a instância da classe *SeedTable* comunica ao objeto oriundo da classe *AuthNetwork* a realização da operação. A aplicação então retorna ao usuário uma mensagem comunicando a realização da exclusão.

O diagrama de seqüência representando este caso de uso está apresentado na Figura 4.7.

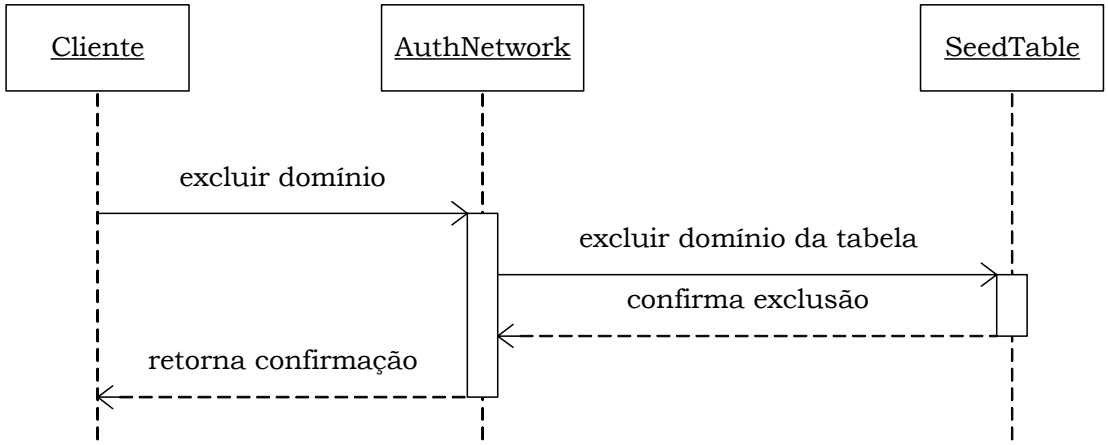

Figura 4.7: Exclusão de um domínio virtual.

4.6.1.5 Inclusão de um domínio virtual sem definição explícita da semente

O usuário pode incluir um novo domínio virtual na sua relação de domínios virtuais conhecidos através apenas da definição de um nome para este domínio, sem que seja necessária a escolha de uma semente a ser utilizado pelo PRNG, que é produzida automaticamente pelo mecanismo.

Assim sendo, o objeto instanciado da classe *AuthNetwork* executa um método estático da classe *PRNG* para a produção de uma nova semente, utilizando como parâmetros o nome do domínio virtual definido pelo usuário e o valor do tempo corrente dado em milésimos de segundos, de acordo com o padrão UTC.

O método estático da classe *PRNG* comunica então o valor randômico a ser utilizado como semente para o objeto oriundo da classe *AuthNetwork* que executa um método do objeto instanciado da classe *SeedTable* que, utilizando o nome do domínio virtual e o valor da semente previamente gerada pelo método estático da classe *PRNG* como parâmetros, executa o armazenamento da semente.

Ao final do processo, o objeto instanciado da classe *SeedTable* comunica ao objeto da classe *AuthNetwork* a realização da operação. A aplicação então retorna ao usuário uma mensagem comunicando a realização da inclusão.

O diagrama de seqüência representando este caso de uso está apresentado na Figura 4.8.

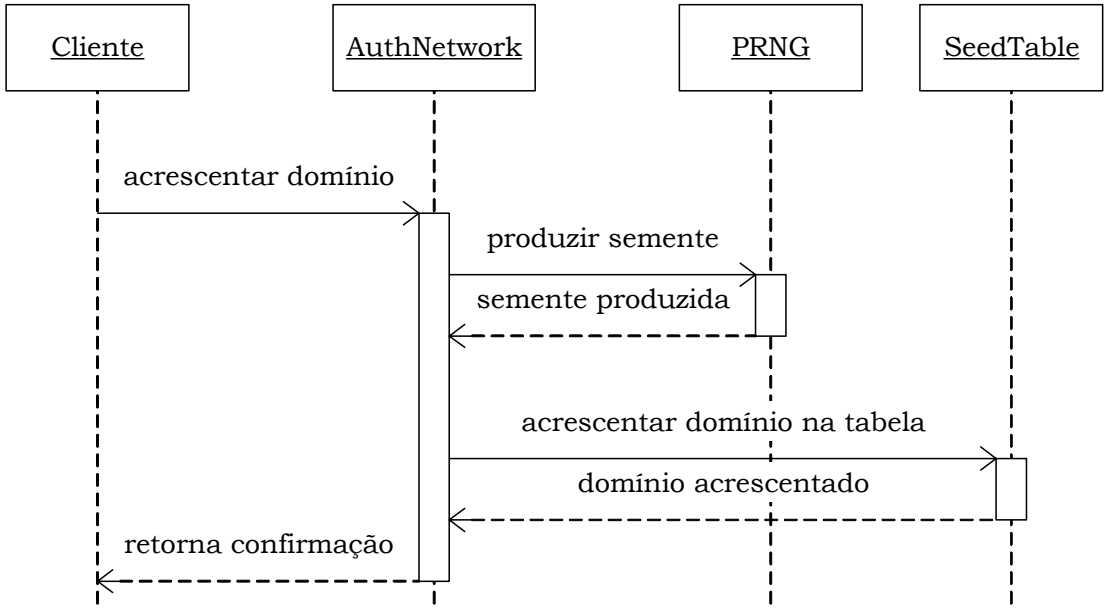

Figura 4.8: Inclusão de um domínio virtual sem definição da semente.

4.6.2 Identificação de um Dispositivo

O usuário pode localizar outros dispositivos pertencentes a um domínio virtual conhecido através da definição do nome do dispositivo a ser verificado e o domínio a ser verificado.

Assim sendo, o objeto instanciado da classe *AuthNetwork* recupera a semente a ser utilizada pelo PRNG através da consulta a um método do objeto oriundo da classe *SeedTable*, tendo como parâmetro o nome do domínio definido pelo usuário. O objeto da classe *SeedTable* retorna a semente recuperada para o objeto da classe *AuthNetwork*, que o utiliza como parâmetro do método estático da classe *PRNG*, usando-a na produção de um valor pseudo-randômico.

Este valor é devolvido para o objeto da classe *AuthNetwork*, que o armazena juntamente com o parâmetro de tempo utilizado na geração do valor pseudo-aleatório, utilizando para isso um método do objeto instanciado da classe *MessageIndex*.

Uma mensagem do tipo *desafio* é enviada pelo objeto da classe *AuthNetwork* do dispositivo do usuário para o objeto da classe *AuthNetwork* do dispositivo submetido a ser identificado, que deve verificar se o conteúdo da mensagem recebida corresponde ao valor esperado para qualquer um de seus domínios virtuais cadastrados.

Esta verificação é feita através da recuperação das sementes dos domínios virtuais cadastrados utilizando um método do objeto instanciado da classe *SeedTable*, que retorna ao objeto da classe *AuthNetwork* uma semente pertencente a um dos domínios virtuais ao qual este dispositivo faz parte.

O objeto da classe *AuthNetwork* faz uso de um método estático da classe *PRNG* que calcula um valor pseudo-randômico, tendo como parâmetros o índice de tempo recebido na mensagem enviada pelo dispositivo do usuário e a semente recuperada do objeto da classe *SeedTable*. O valor pseudo-aleatório produzido é então retornado para o objeto da classe *AuthNetwork* que verifica se os 40 primeiros bits do valor gerado correspondem ao valor recebido na mensagem do tipo *desafio*.

Esta verificação é executada para todas as sementes dos domínios virtuais cadastrados até que se determine o domínio ao qual o valor recebido na mensagem proveniente do dispositivo cliente pertença. Caso esta verificação determine que o valor recebido não tenha sido produzido por nenhum domínio virtual de conhecimento do dispositivo, a comunicação é encerrada.

No entanto, caso esta verificação determine o domínio virtual utilizado na produção da mensagem *desafio* recebida, o objeto da classe *AuthNetwork* produz um segundo valor pseudo-randômico através do uso de um método estático da classe *PRNG*, tendo como parâmetros a mesma semente utilizada para produzir o valor recebido e um novo índice de tempo, sendo todos os valores citados armazenados através de um método do objeto instanciado da classe *MessageIndex*.

Uma mensagem do tipo *resposta* é então enviada para o dispositivo do usuário que, verifica as informações recebidas, repetindo o processo já descrito, e enviando uma mensagem do tipo *réplica* para o segundo

dispositivo, que a analisa, novamente repetindo o processo de verificação que, uma vez bem sucedido, transmite uma mensagem de confirmação para o objeto da classe *AuthNetwork* do dispositivo do usuário.

Ao final deste processo, a aplicação retorna ao usuário uma mensagem comunicando o sucesso do processo de identificação do dispositivo em questão.

O diagrama de seqüência apresentado na Figura 4.9 representa uma simplificação deste caso de uso, eliminando algumas passagens que já tenham sido descritas em algum ponto do próprio diagrama como, por exemplo, o processo de geração do segundo valor pseudo-randômico executado pelo segundo dispositivo (dispositivo *B* na Figura 4.9), que é idêntico ao processo apresentado para o primeiro dispositivo (dispositivo *A* na Figura 4.9).

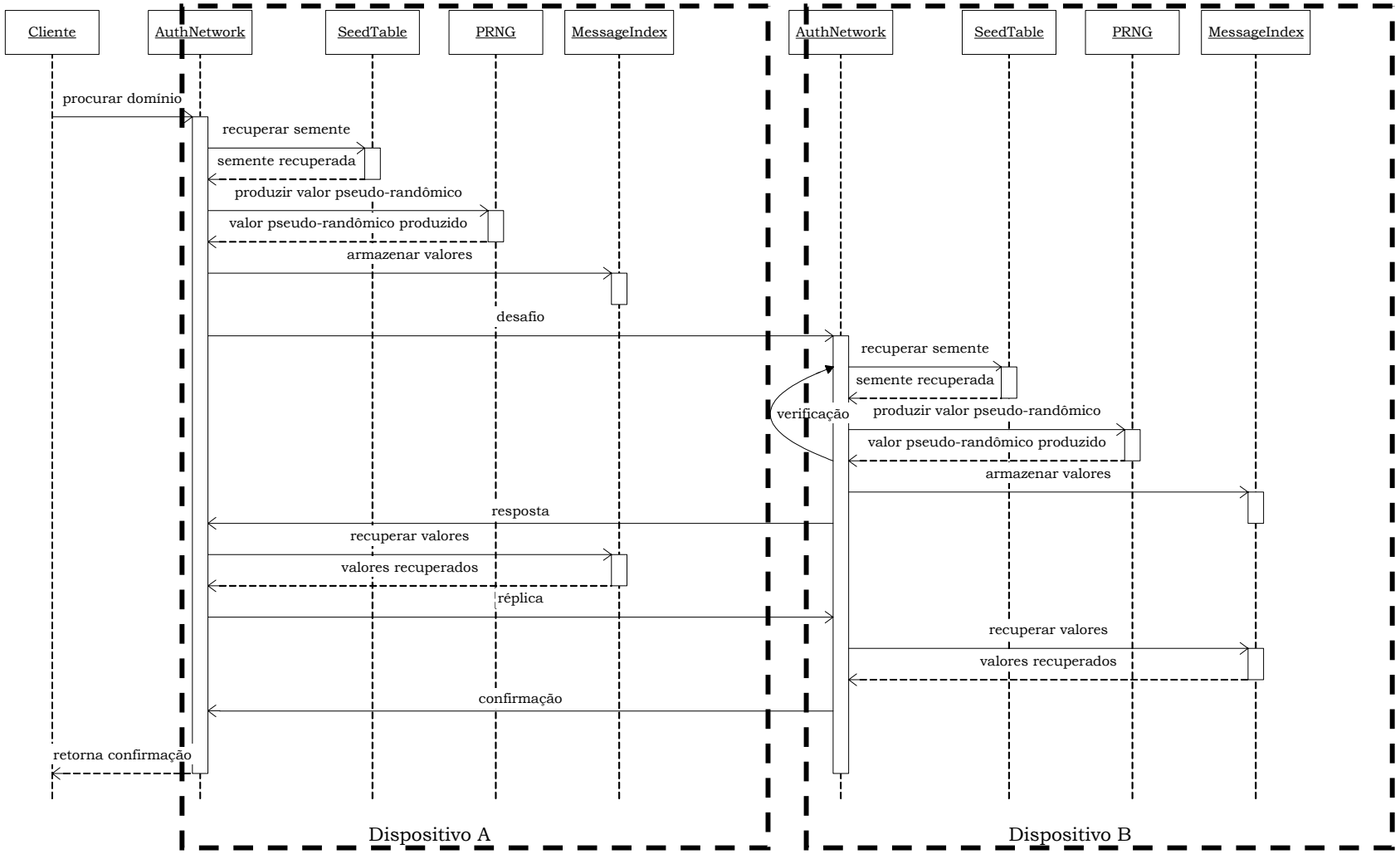

Figura 4.9: Diagrama de seqüência simplificado do processo de identificação de um dispositivo.

4.7 Considerações sobre a Implementação

A implementação do protótipo do mecanismo de Domínios Virtuais, apresentado ao longo deste capítulo, permite que um dispositivo pertencente a um determinado domínio virtual possa identificar outro dispositivo que também pertença a este mesmo domínio.

Deve-se ressaltar que o protótipo é uma aplicação desenvolvida com o propósito único de demonstrar a viabilidade da implementação e o funcionamento do mecanismo de Domínios Virtuais proposto nesta dissertação. Assim sendo, a primeira versão do protótipo não inclui o armazenamento persistente das mensagens recebidas pelo dispositivo, sendo estas descartadas ao final do processo.

Para que este protótipo possa ser utilizado como um mecanismo de segurança, o mesmo deve estar posicionado entre as aplicações e a comunicação de dados, como apresentado previamente na Figura 3.2, de modo que qualquer comunicação de dados seja antes submetida ao processo de identificação imposto pelo mecanismo de Domínios Virtuais.

Inicialmente, o protótipo foi posicionado como apresentado na Figura 4.10, ou seja, paralelamente às demais aplicações.

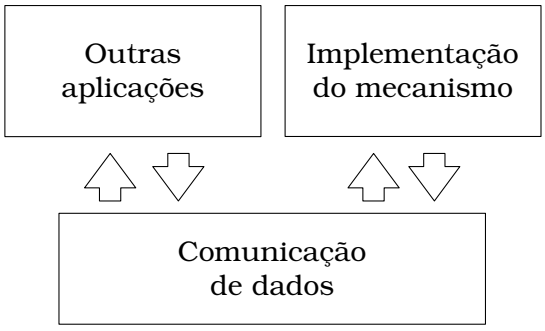

Figura 4.10: Posicionamento do protótipo desenvolvido.

A aplicação desenvolvida foi corretamente posicionada entre as aplicações e o bloco de comunicação de dados, de acordo com a Figura 3.2, quando as classes que a implementam foram adicionadas à arquitetura de segurança do qual este mecanismo é parte integrante. Assim sendo, foram criadas condições para que a implementação do mecanismo de Domínios Virtuais

fosse aplicada de acordo com sua especificação, a assim, definitivamente colaborar com o desenvolvimento de um ambiente móvel seguro.

Capítulo 5

Testes e Resultados

"Anyone who considers arithmetical methods of producing random digits is, of course, in a state of sin."

John von Neumann

A descrição da metodologia de testes utilizada para a avaliação da implementação do mecanismo de Domínios Virtuais, assim como a apresentação dos testes efetuados seguindo a metodologia apresentada correspondem aos objetivos deste capítulo.

Deste modo, este capítulo inicialmente apresenta uma metodologia de testes a ser seguida para a avaliação do mecanismo, na qual propõe a divisão dos testes de acordo com os diversos blocos funcionais que compõem a implementação, de modo a testá-los separadamente para que, só então, um teste final possa ser realizado, tendo então suas diversas partes integradas.

A seguir, é dada atenção especial ao PRNG, já que este corresponde a uma variação de um padrão existente, tornando necessária sua avaliação. Os testes seguem com a avaliação individual das classes que compõem a implementação, para que, finalmente, a aplicação possa ser testada.

5.1 Metodologia de Testes Utilizada

Uma metodologia[48] de testes a ser aplicada sobre a aplicação que implementa o mecanismo de Domínios Virtuais, proposto nesta dissertação, se faz necessária para que seja possível alcançar uma avaliação isenta da

[48] **metodologia.** [Do gr. *méthodos*, 'método', + *-log(o)-* + *ia.*] *S.f.* **1**.A arte de dirigir o espírito na arte da investigação. **2**. *Liter.* Conjunto de técnicas e processos utilizados para ultrapassar a subjetividade do autor e atingir a obra literária.
FERREIRA, Aurélio Buarque de Holanda. Novo Dicionário da Língua Portuguesa. 2ª edição. Rio de Janeiro.

subjetividade do avaliador, e também passível de ser repetida sempre que necessário.

A metodologia consiste basicamente na divisão da aplicação em seus diversos blocos funcionais, de modo que cada um possa ser avaliado individualmente de acordo com suas funcionalidades. Após a conclusão da verificação de seus blocos funcionais, a aplicação é finalmente avaliada, testando-se assim o funcionamento conjunto de suas partes constituintes e assim, permitindo que seja possível alcançar uma conclusão final sobre a implementação efetuada.

A aplicação foi dividida em três blocos principais:

- O gerador de números aleatórios é o primeiro bloco a ser avaliado. Os testes têm o objetivo de detectar fraquezas no PRNG através de análise estatística da distribuição de valores gerados. Além disso, o PRNG é submetido a uma bateria de teste padrão, a FIPS PUB 140-2 [17] e a outros testes individuais.
- Operações sobre informações armazenadas em tabelas são testadas de modo a avaliar o correto funcionamento das mesmas. Os testes executados verificam se os dados são manipulados corretamente nestas operações.
- Reconhecimento de um domínio virtual. Este teste corresponde à avaliação final do mecanismo, no qual todas as partes são integradas, sendo então possível verificar o funcionamento correto dos processos de tratamento, transmissão, e recepção de mensagens, além de mecanismos adicionais de segurança, como as janelas de tempo.

5.2 O PRNG do Mecanismo de Domínios Virtuais

O mecanismo de Domínios Virtuais, proposto nesta dissertação, é fundamentado na existência de um gerador de números pseudo-aleatórios, presente nos diversos dispositivos que compõem uma rede móvel ad hoc, que possua propriedades criptográficas, que seja capaz de aceitar um

parâmetro de entrada que possa ser considerado de conhecimento público e que seja adequado para dispositivos móveis, de modo a respeitar suas limitações, principalmente em relação à quantidade de processamento exigida e, conseqüentemente, ao consumo de energia requerido.

O estudo de viabilidade dos PRNG candidatos, descrito no item 3.3.1, indicou que o único PRNG que poderia ser utilizado sem que existisse nenhuma modificação em seu algoritmo, o Blum Blum Shub, é computacionalmente intensivo, o que o levou a ser descartado, restando apenas dois outros PRNG que poderiam ser utilizados, após serem devidamente modificados, o ANSI X9.31 e o DSA. Finalmente, após a comparação de ambos, apresentada no item 3.3.1.6, optou-se pela utilização do PRNG DSA modificado.

O novo PRNG, obtido da modificação do DSA, deve ser submetido a uma avaliação de suas propriedades, de modo a garantir que a alteração efetuada em seu algoritmo não tenha causado o comprometimento das propriedades do mesmo.

É importante ressaltar que não é possível alcançar a comprovação matemática de que o novo gerador é, de fato, um PRNG, assim como é impossível prová-lo para qualquer outro gerador existente. Os testes a serem executados têm o objetivo único de detectar a existência de possíveis fraquezas no gerador, sem que, no entanto, possa-se determinar que se trata verdadeiramente de um PRNG [75].

Deste modo, um resultado positivo nos testes, a serem apresentados neste capítulo, é condição necessária, mas não suficiente, para qualquer gerador candidato a PRNG, de modo que as conclusões obtidas não podem ser consideradas definitivas, mas sim, probabilísticas [74] [75].

5.2.1 Produção de Amostras

Os testes foram realizados a partir de amostras obtidas do gerador em questão. Foram produzidas então quatro amostras de cem mil valores de comprimento e outras quatro de dez mil valores de comprimento, utilizando-se para isso diferentes sementes e também o incremento artificial

do índice de tempo, através de acréscimos lineares de um milésimo de segundo para as maiores amostras e de um segundo para as menores.

As amostras foram então nomeadas e são apresentadas na Tabela 5.1.

Tabela 5.1: Amostras produzidas.

Nome	Tamanho da amostra.(10^{-3})
Escritório	10
Lar doce Lar	10
Litoral	10
Casa de Campo	10
Donovan	100
Journey	100
Supertramp	100
Grassroots	100

Na qual cada unidade da amostra possui 160bits de comprimento e sua representação é dada em complemento de dois.

5.2.2 Teste χ^2

De modo a avaliar se as amostras produzidas no gerador podem ser consideradas identicamente e independentemente distribuídas (IID)$U(0,1)$, estas foram submetidas a um teste de hipótese não-paramétrico, no qual é avaliado se a distribuição observada adere à uma distribuição χ^2 [75] [76].

Histogramas foram construídos a partir das amostras e submetidos ao teste de aderência χ^2, que consiste no resultado na seguinte equação:

$$D = \chi_v^2 = \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i} \tag{14}$$

Sendo que:

- O_i é a freqüência observada de um determinado intervalo.
- E_i é a freqüência esperada para um determinado intervalo.
- k é o número total de intervalos considerados.

- v corresponde ao número de graus de liberdade da distribuição χ^2, sendo que:

$$v = k - 1 \tag{15}$$

O nível de significância α adotado é de 0.05[49].

5.2.2.1 Testes das amostras de 10^4 valores

As amostras de 10^4 valores foram divididas em 51 intervalos, de modo que a sua distribuição deve aderir a uma distribuição χ^2 com 50 graus de liberdade. Deste modo, podemos verificar na tabela de distribuições χ^2 que temos para um nível de significância α igual a 0.05, e v igual a 50:

$$\chi^2_{50;0,05} = 67,505 \tag{16}$$

A freqüência esperada E_i é igual ao número total de amostras dividido pela quantidade de intervalos, ou seja, cerca de 196 unidades são esperadas em cada célula. Deste modo, para todas as amostras de 10^4 valores:

$$E_i = 196 \tag{17}$$

[49] Para a avaliação em questão, o nível de significância adotado pode variar entre 0.001 e 0.05 [74].

Escritório

Para a amostra *Escritório*, foi obtida a distribuição apresentada na Figura 5.1.

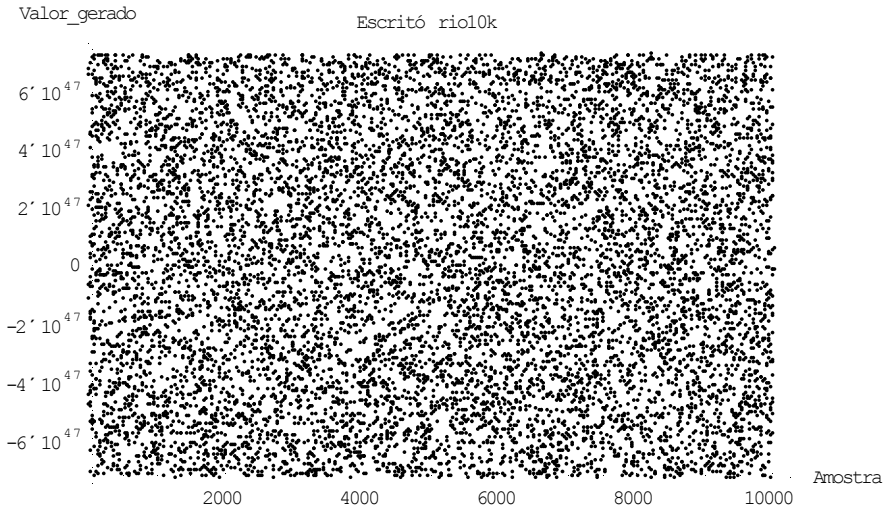

Figura 5.1: Distribuição da amostra *Escritório*.

O histograma produzido a partir desta distribuição está apresentado na Figura 5.2.

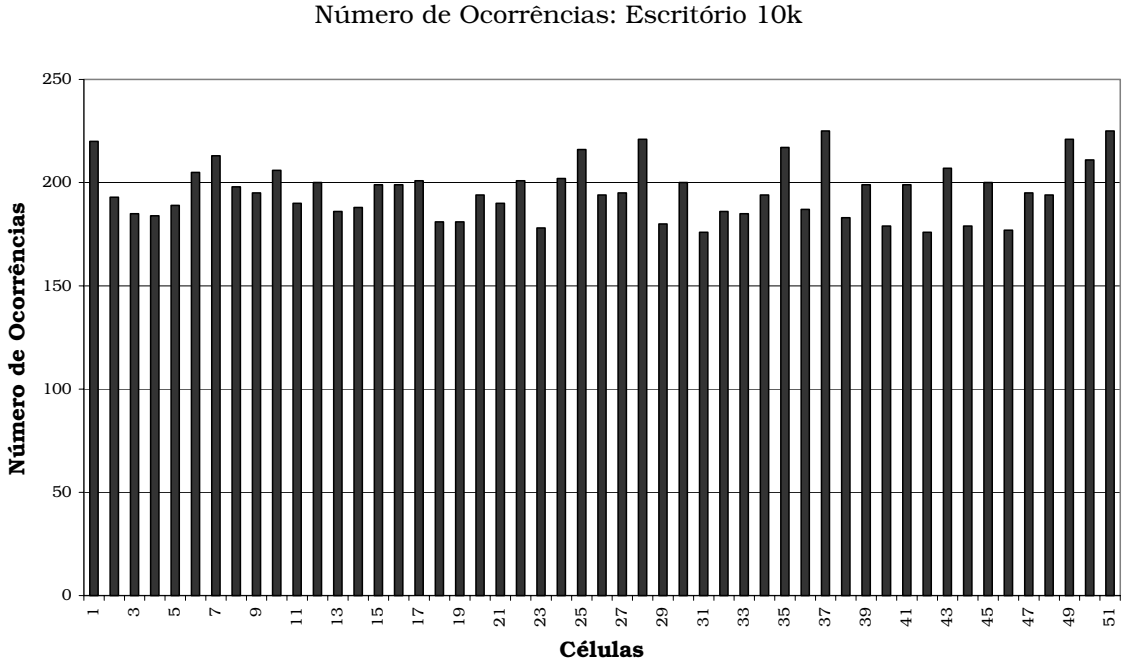

Figura 5.2: Histograma obtido da amostra *Escritório*.

O cálculo do valor da estatística de teste é obtido através da aplicação da equação (14), tendo seu parâmetro E_i dado na equação (17). Sendo assim, para a amostra *Escritório* de 10^4 valores, temos que:

$$D_{Escritório} = 46{,}704 \tag{18}$$

Satisfazendo, portanto, a condição para aceitação da amostra ao nível de significância α igual a 0.05, já que:

$$D_{Escritório} < \chi^2_{50;0,05} \tag{19}$$

Lar doce Lar

Para a amostra *Lar doce Lar*, foi obtida a distribuição apresentada na Figura 5.3.

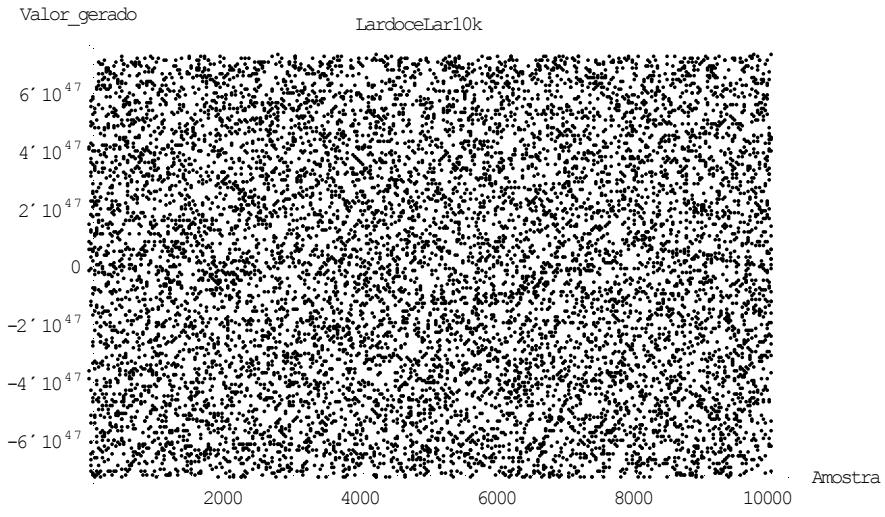

Figura 5.3: Distribuição da amostra *Lar doce Lar*.

O histograma produzido a partir desta distribuição está apresentado na Figura 5.4.

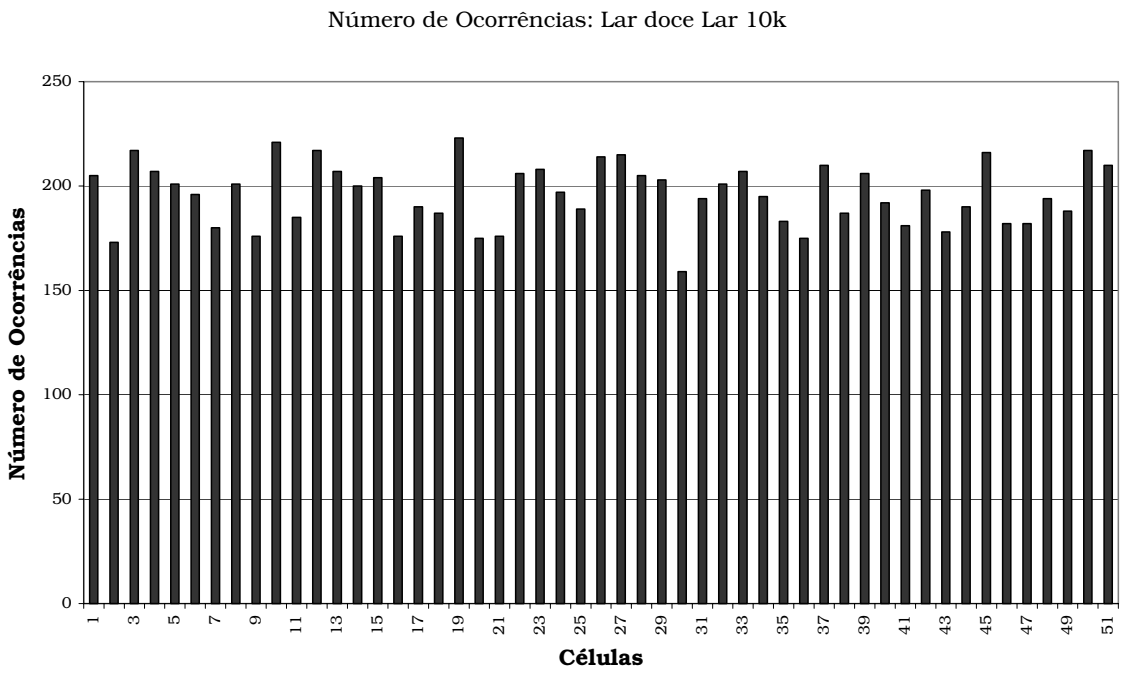

Figura 5.4: Histograma obtido da amostra *Lar doce Lar*.

Novamente, o cálculo do valor da estatística de teste é obtido através da aplicação da equação (14), tendo seu parâmetro E_i dado na equação (17). Sendo assim, para a amostra *Lar doce Lar* de 10k valores, temos que:

$$D_{Lar\_doce\_Lar} = 56{,}467 \tag{20}$$

Satisfazendo, portanto, a condição para aceitação da amostra ao nível de significância α igual a 0.05, já que:

$$D_{Lar\_doce\_Lar} < \chi^2_{50;0,05} \tag{21}$$

Litoral

Para a amostra *Litoral*, foi obtida a distribuição apresentada na Figura 5.5.

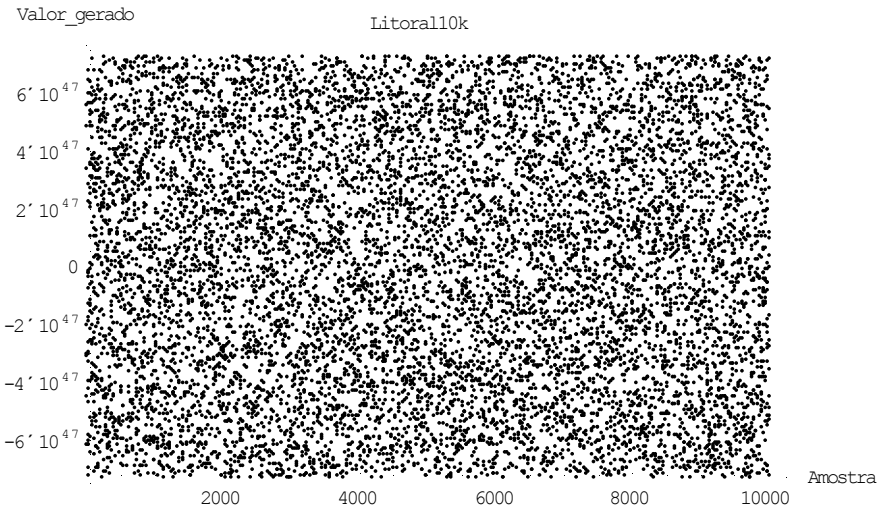

Figura 5.5: Distribuição da amostra *Litoral*.

O histograma produzido a partir desta distribuição está apresentado na Figura 5.6.

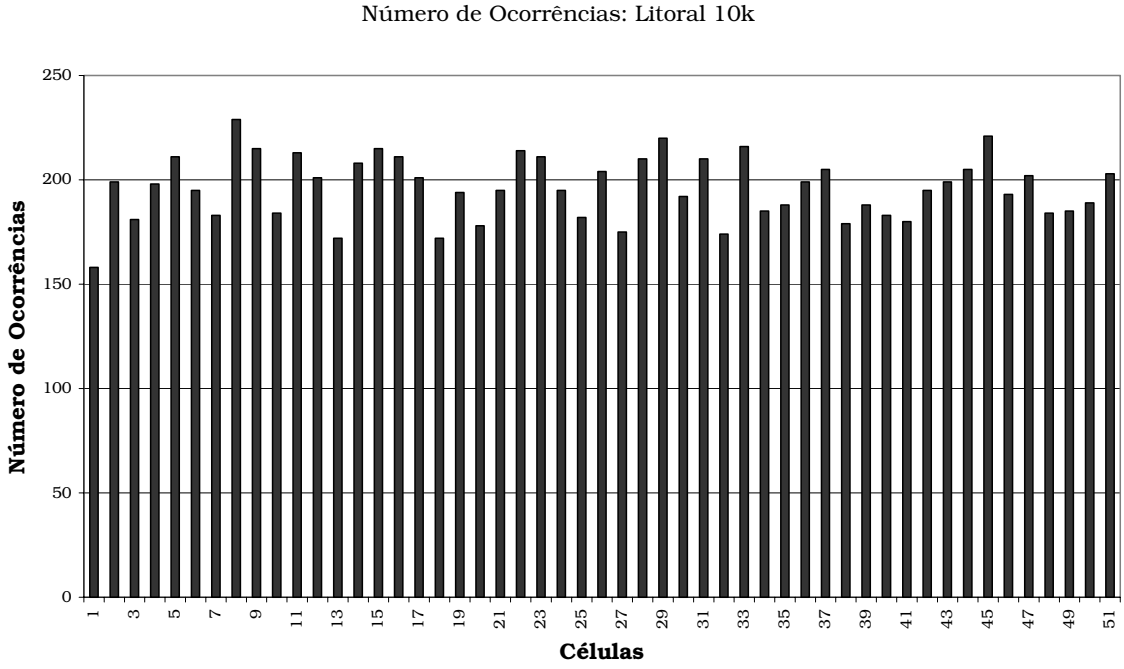

Figura 5.6: Histograma obtido da amostra *Litoral*.

Novamente, o cálculo do valor da estatística de teste é obtido através da aplicação da equação (14), tendo seu parâmetro E_i dado na equação (17). Sendo assim, para a amostra *Litoral* de 10^4 valores, temos que:

$$D_{Litoral} = 58{,}742 \tag{22}$$

Satisfazendo, portanto, a condição para aceitação da amostra ao nível de significância α igual a 0.05, já que:

$$D_{Litoral} < \chi^2_{50;0,05} \tag{23}$$

Casa de Campo

A seguinte distribuição, apresentada na Figura 5.7, foi obtida da amostra *Casa de Campo*.

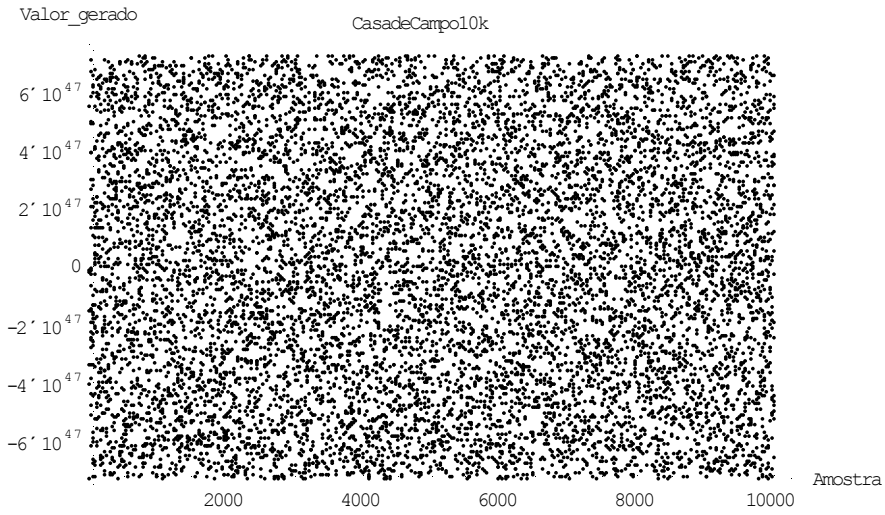

Figura 5.7: Distribuição da amostra *Casa de Campo*.

O histograma produzido a partir desta distribuição está apresentado na Figura 5.8.

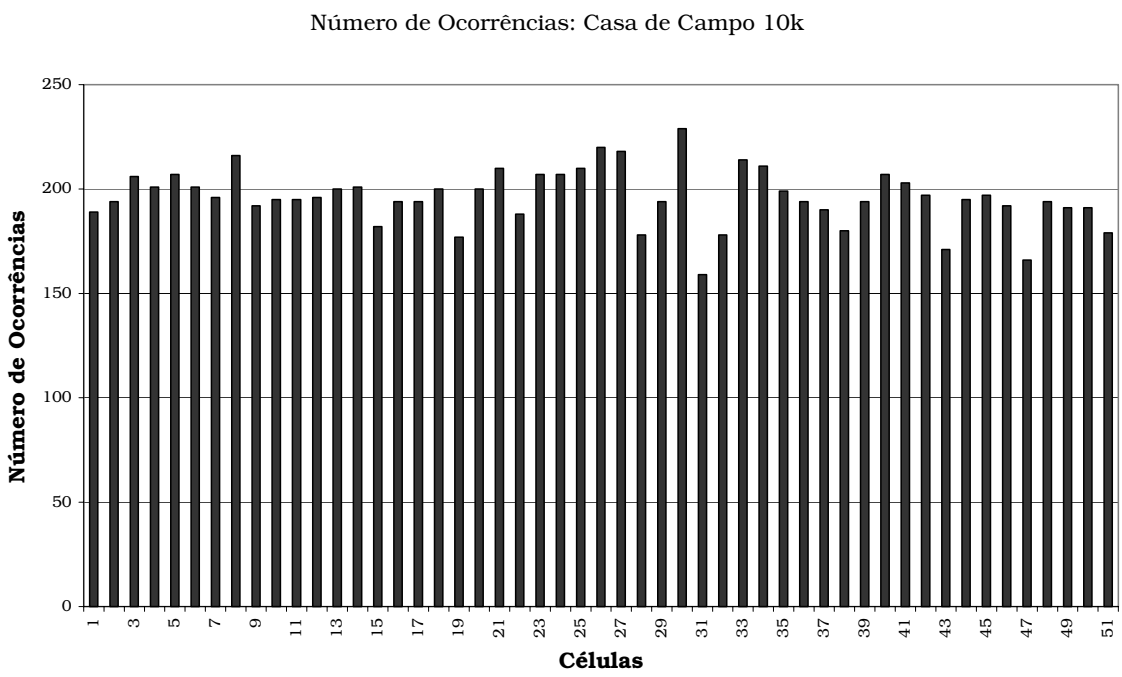

Figura 5.8: Histograma obtido da amostra *Casa de Campo*.

Novamente, o cálculo do valor da estatística de teste é obtido através da aplicação da equação (14), tendo seu parâmetro E_i dado na equação (17). Sendo assim, para a amostra *Casa de Campo* de 10^4 valores, temos que:

$$D_{Casa\_de\_Campo} = 46{,}735 \tag{24}$$

Satisfazendo, portanto, as condição para aceitação da amostra ao nível de significância α igual a 0.05, já que:

$$D_{Casa\_de\_Campo} < \chi^2_{50;0,05} \tag{25}$$

5.2.2.2 Testes das amostras de 100k valores

As amostras de 100k valores foram divididas em intervalos de 51 unidades, de modo que a sua distribuição deve aderir a uma distribuição χ^2 com 50 graus de liberdade. Novamente, podemos verificar na tabela de distribuições χ^2 que temos para um nível de significância α igual a 0.05, e υ igual a 50:

$$\chi^2_{50;0,05} = 67{,}505 \tag{26}$$

A freqüência esperada E_i é igual ao número total de amostras dividido pela quantidade de intervalos, ou seja, cerca de 1961 unidades são esperadas em cada célula. Deste modo, para todas as amostras de 10^5 valores:

$$E_i = 1961 \tag{27}$$

A representação das distribuições encontradas foi omitida neste item, pois a utilização de amostras contendo 10^5 valores acarreta uma alta densidade de pontos nos gráficos, tornando a visualização dos mesmos inviável, e sua apresentação, portanto, inútil, já que não fornecem informações relevantes.

Donovan

O histograma produzido a partir da distribuição produzida por esta amostra está apresentado na Figura 5.9.

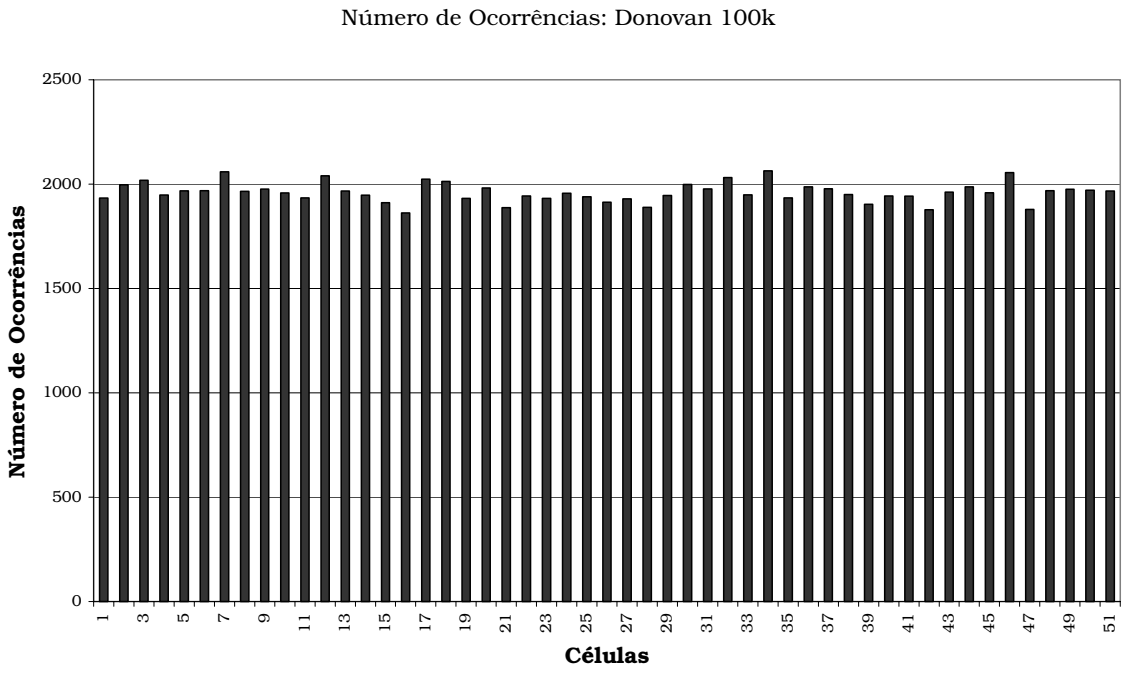

Figura 5.9: Histograma obtido da amostra *Donovan*.

O cálculo do valor da estatística de teste é obtido através da aplicação da equação (14), tendo seu parâmetro E_i dado na equação (17). Sendo assim, para a amostra *Donovan* de 10^5 valores, temos que:

$$D_{Donovan} = 53{,}796 \tag{28}$$

Satisfazendo, portanto, a condição para aceitação da amostra ao nível de significância α igual a 0.05, já que:

$$D_{Donovan} < \chi^2_{50;0,05} \tag{29}$$

Journey

O histograma produzido a partir da distribuição produzida por esta amostra está apresentado na Figura 5.10.

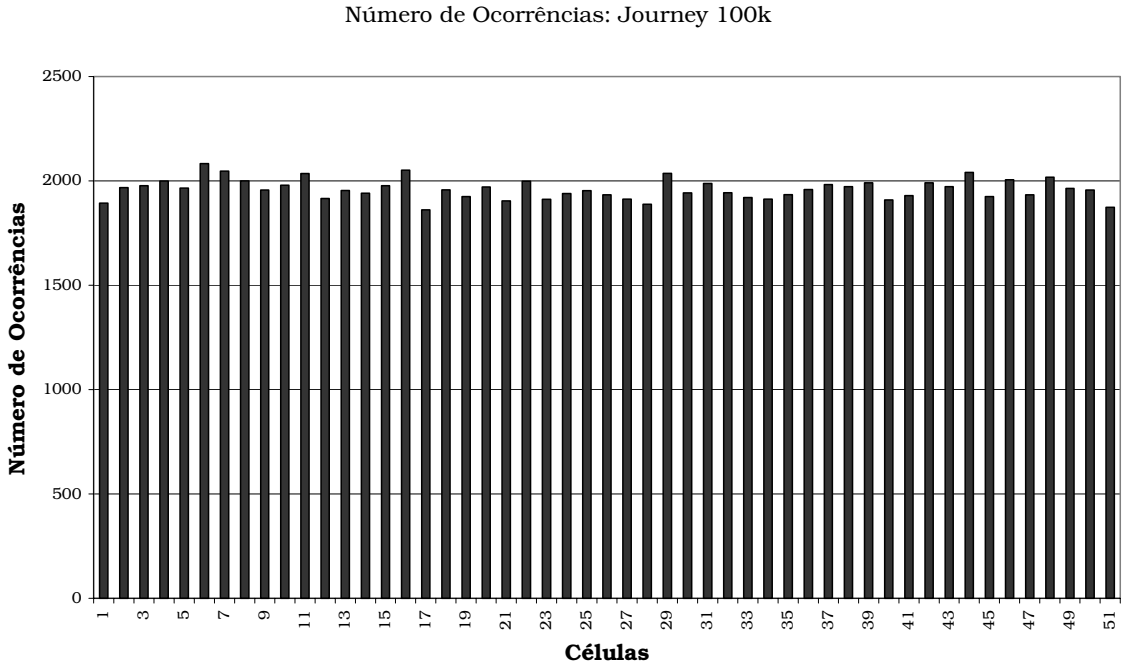

Figura 5.10: Histograma obtido da amostra *Journey*.

O cálculo do valor da estatística de teste é obtido através da aplicação da equação (14), tendo seu parâmetro E_i dado na equação (17). Sendo assim, para a amostra *Journey* de 10^5 valores, temos que:

$$D_{Journey} = 57{,}996 \tag{30}$$

Satisfazendo, portanto, a condição para aceitação da amostra ao nível de significância α igual a 0.05, já que:

$$D_{Journey} < \chi^2_{50;0,05} \tag{31}$$

Supertramp

O histograma produzido a partir da distribuição produzida por esta amostra está apresentado na Figura 5.11.

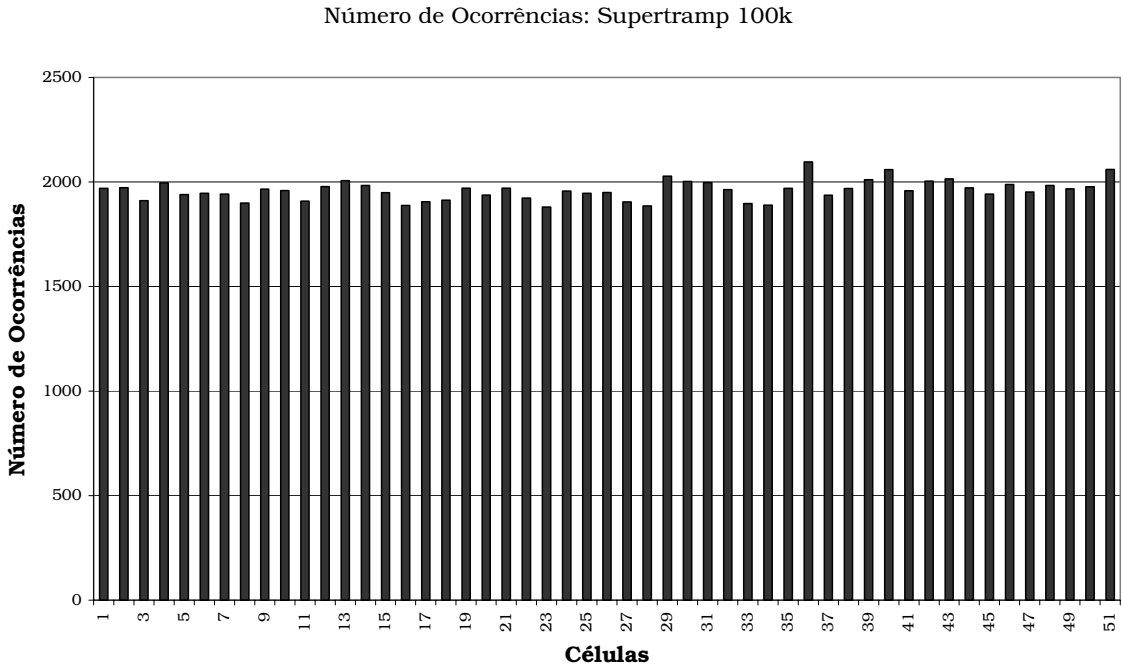

Figura 5.11: Histograma obtido da amostra *Supertramp*.

O cálculo do valor da estatística de teste é obtido através da aplicação da equação (14), tendo seu parâmetro E_i dado na equação (17). Sendo assim, para a amostra *Supertramp* de 10^5 valores, temos que:

$$D_{Supertramp} = 55{,}065 \tag{32}$$

Satisfazendo, portanto, a condição para aceitação da amostra ao nível de significância α igual a 0.05, já que:

$$D_{Supertramp} < \chi^2_{50;0,05} \tag{33}$$

Grassroots

O histograma produzido a partir da distribuição produzida por esta amostra está apresentado na Figura 5.12.

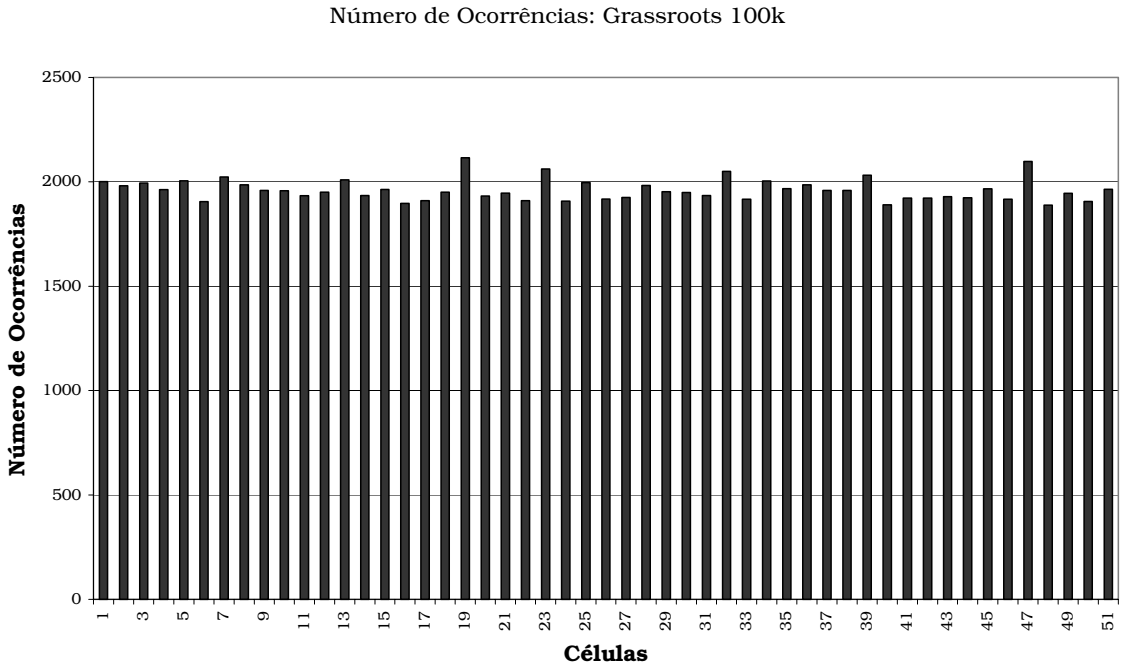

Figura 5.12: Histograma obtido da amostra *Grassroots*.

O cálculo do valor da estatística de teste é obtido através da aplicação da equação (14), tendo seu parâmetro E_i dado na equação (17). Sendo assim, para a amostra *Grassroots* de 10^5 valores, temos que:

$$D_{Grassroots} = 64{,}900 \tag{34}$$

Satisfazendo, portanto, a condição para aceitação da amostra ao nível de significância α igual a 0.05, já que:

$$D_{Grassroots} < \chi^2_{50;0,05} \tag{35}$$

5.2.2.3 Conclusões sobre os resultados dos testes χ^2

Com os resultados obtidos nos testes de hipótese executados nos itens 5.2.2.1 e 5.2.2.2, pode-se afirmar que, ao nível de significância α igual a 0.05, que todas as amostras aderem a uma distribuição χ^2, de modo que

pode-se aceitar os valores produzidos pelo gerador utilizado como (IID)$U(0,1)$.

A Tabela 5.2, abaixo, apresenta os resultados obtidos neste item.

Tabela 5.2: Resultados obtidos.

Nome	$D_{calculado}$
Escritório	46,704
Lar doce Lar	56,467
Litoral	58,742
Casa de Campo	46,735
Donovan	53,796
Journey	57,996
Supertramp	55,065
Grassroots	64,900

5.2.3 Outros Testes

Apesar de ter sido bem sucedido no teste χ^2, a utilização de apenas um tipo de avaliação não é, no entanto, suficiente para a verificação de um gerador, tornando-se necessária a aplicação de outros testes [75].

A seguir, quatro testes são aplicados a uma única seqüência de 2.10^4bits produzida pelo gerador. Esta seqüência foi montada através da concatenação de 125 valores pseudo-randômicos de 160bits gerados seqüencialmente.

As quatro seqüências de 2.10^4bits produzidas foram nomeadas de acordo com a Tabela 5.3:

Tabela 5.3: Nome das amostras de 2.10^4bits.

Nome da amostra
Escritório
Lar doce Lar
Litoral
Casa de Campo

Utilizando estas amostras, os testes a seguir foram realizados.

5.2.3.1 Teste de freqüência (monobit)

O objetivo deste teste é determinar se a quantidade de zeros e uns encontrados na seqüência é aproximadamente o mesmo, o que é naturalmente esperado para uma seqüência aleatória. O teste aplicado é apresentado na equação (36) [74].

$$X_1 = \frac{(n_0 - n_1)^2}{n} \tag{36}$$

Sendo que:

- n é o comprimento total da amostra, ou seja, 2.10^4.
- n_0 corresponde à quantidade de zeros da amostra.
- n_1 corresponde à quantidade de uns da amostra.

O resultado obtido na equação (36) deve aderir a uma distribuição χ^2, com um grau de liberdade. O nível de significância α adotado é, novamente, igual a 0.05.

$$\chi^2_{1;0,05} = 3,841 \tag{37}$$

Deste modo, aplicando-se a equação (36) para cada uma das amostras, obtém-se os resultados apresentados na Tabela 5.4.

Tabela 5.4: Resultados do teste de freqüência.

Nome da amostra	X_1	$X_1 < \chi^2_{1;0,05}$
Escritório	0,168	Verdadeiro
Lar doce Lar	0,405	Verdadeiro
Litoral	2,789	Verdadeiro
Casa de Campo	0,106	Verdadeiro

Com os resultados apresentados na Tabela 5.4 é possível afirmar que, ao nível de significância α igual a 0.05, todas as amostras aderem a uma distribuição χ^2, ou seja, passam no teste de freqüência.

5.2.3.2 Teste serial (two-bit test)

O objetivo deste teste é verificar se a ocorrência dos quatro possíveis pares de bits, 00, 01, 10 e 11, acontece aproximadamente com a mesma freqüência, o que é naturalmente esperado em uma verdadeira seqüência randômica. O teste aplicado é apresentado na equação (38) [74].

$$X_2 = \frac{4}{n-1} \cdot \left(n_{00}^2 + n_{01}^2 + n_{10}^2 + n_{11}^2\right) - \frac{2}{n} \cdot \left(n_0^2 + n_1^2\right) + 1 \tag{38}$$

Sendo que:

- n é o comprimento total da amostra, ou seja, 2.10^4.
- n_0 corresponde à quantidade de zeros da amostra.
- n_1 corresponde à quantidade de uns da amostra.
- n_{00} corresponde à quantidade de pares 00 da amostra.
- n_{01} corresponde à quantidade de pares 01 da amostra.
- n_{10} corresponde à quantidade de pares 10 da amostra.
- n_{11} corresponde à quantidade de pares 11 da amostra.

Os pares são gerados utilizando-se a sobreposição dos bits da seqüência, sendo que o primeiro par é construído através da concatenação do primeiro e do segundo bit, enquanto que o segundo par é composto pelo segundo e terceiro bit, de modo que:

$$n_{00} + n_{01} + n_{10} + n_{11} = n - 1 \tag{39}$$

O resultado obtido na equação (38) deve aderir a uma distribuição χ^2, com dois graus de liberdade. O nível de significância α adotado é, novamente, igual a 0.05.

$$\chi^2_{2;0,05} = 5,991 \tag{40}$$

Deste modo, através da aplicação da equação (38) para cada uma das amostras, obtêm-se os resultados apresentados na Tabela 5.5.

Tabela 5.5: Resultados do teste serial.

Nome da amostra	X_2	$X_2 < \chi^2_{2;0,05}$
Escritório	0,929	Verdadeiro
Lar doce Lar	2,451	Verdadeiro
Litoral	1,953	Verdadeiro
Casa de Campo	1,815	Verdadeiro

Com os resultados apresentados na Tabela 5.5 é possível afirmar que, ao nível de significância α igual a 0.05, todas as amostras aderem a uma distribuição χ^2, ou seja, passam no teste serial.

5.2.3.3 Teste poker (poker test)

O objetivo deste teste é verificar se cadeias de bits de comprimento *m*, retirados da amostra, ocorrem aproximadamente o mesmo número de vezes, como é característico de um valor randômico. Considerando-se que *m* é um inteiro positivo e obedece a condição imposta na equação (41), abaixo [74]:

$$\left[\frac{n}{m}\right] \geq 5 \cdot \left(2^m\right) \tag{41}$$

E que a amostra contendo *n* elementos é dividida em *k* seqüências sem que ocorra sobreposição de bits, de modo que:

$$k = \left[\frac{n}{m}\right] \tag{42}$$

Aplicando-se então o seguinte teste estatístico, apresentado na equação (43).

$$X_3 = \frac{2^m}{k} \cdot \left(\sum_{i=1}^{2^m} n_i^2\right) - k \tag{43}$$

Sendo que:

- n_i corresponde ao número de ocorrências do *i*-ésimo tipo de seqüência de comprimento *m*, e:

$$1 \le i \le 2^m \tag{44}$$

O resultado obtido através da aplicação da equação (43) deve aderir a uma distribuição χ^2, com (2^m-1) graus de liberdade, de modo que se pode assumir que este teste é uma generalização do teste de freqüência [74].

O nível de significância α adotado é, novamente, igual a 0.05 e os testes foram efetuados adotando-se *m* igual a 4, existindo, portanto, 2^4 possíveis seqüências de bits e assumindo-se uma distribuição χ^2 com 15 graus de liberdade, tem-se que:

$$\chi^2_{15;0,05} = 24{,}996 \tag{45}$$

Deste modo, assumindo-se *m* igual a 4 e aplicando-se a equação (43) à amostra de 2.10^4bits produzidos pelo gerador, obtém-se os resultados apresentados na Tabela 5.6.

Tabela 5.6: Resultados dos teste poker.

Nome da amostra	X_3	$X_3 < \chi^2_{15;0,05}$
Escritório	12,781	Verdadeiro
Lar doce Lar	11,750	Verdadeiro
Litoral	13,421	Verdadeiro
Casa de Campo	16,755	Verdadeiro

Com os resultados apresentados na Tabela 5.6 é possível afirmar que, ao nível de significância α igual a 0.05, todas as amostras aderem a uma distribuição χ^2, ou seja, passam no teste poker.

5.2.3.4 Teste de comprimento de seqüências (runs test)

O objetivo deste teste é verificar se o comprimento dos blocos e das lacunas encontrados na seqüência produzida pelo gerador corresponde ao esperado

para um valor randômico, no qual blocos são seqüências ininterruptas de bits um enquanto que lacunas são seqüências ininterruptas de bits zero, podendo blocos e lacunas possuir comprimento arbitrário *i*, sendo i dado na relação apresentada na equação (46).

$$1 \leq i \leq \textit{comprimento do maior bloco ou lacuna} \tag{46}$$

A quantidade de blocos ou lacunas de comprimento *i* esperadas para um valor randômico é dado pela equação (47), na qual *n* representa a quantidade total de elementos da seqüência produzida pelo gerador.

$$e_i = \frac{(n-i+3)}{2^{i+2}} \tag{47}$$

O teste a ser aplicado é apresentado na equação (48).

$$X_4 = \sum_{i=1}^{k} \frac{(B_i - e_i)^2}{e_i} + \sum_{i=1}^{k} \frac{(G_i - e_i)^2}{e_i} \tag{48}$$

Sendo que:

- *k* corresponde ao maior inteiro *i* para o qual $e_i \geq 5$.
- B_i corresponde ao número de ocorrências de blocos de comprimento *i*.
- G_i corresponde ao número de ocorrências de lacunas de comprimento *i*.

O resultado obtido através da aplicação da equação (48) deve aderir a uma distribuição χ^2, com (2*k*-2) graus de liberdade [74].

O índice *k*, a ser aplicado na equação (48), pode ser determinado através da utilização da equação (47), de modo a se detectar qual o primeiro índice e_i que satisfaz *k*. Assim, como:

$$e_9 = 9{,}8 \geq 5 \text{ e } e_{10} = 4{,}9 \leq 5, \text{ tem-se } k = 9 \tag{49}$$

Assim sendo, para o nível de significância α adotado igual a 0.05, temos:

$$\chi^2_{16;0,05} = 26,296 \quad (50)$$

Deste modo, através da aplicação da equação (48) para cada uma das amostras, obtém-se os resultados apresentados na Tabela 5.7.

Tabela 5.7: Resultados do teste de comprimento de seqüência.

Nome da amostra	X_4	$X_4 < \chi^2_{16;0,05}$
Escritório	17,984	Verdadeiro
Lar doce Lar	20,741	Verdadeiro
Litoral	19,769	Verdadeiro
Casa de Campo	14,972	Verdadeiro

Com os resultados apresentados na Tabela 5.7 é possível afirmar que, ao nível de significância α igual a 0.05, todas as amostras aderem a uma distribuição χ^2, ou seja, passam no teste de comprimento de seqüência.

5.2.3.5 Conclusões sobre os testes realizados

Todos os testes levantados neste item (de freqüência, serial, *poker* e de comprimento de seqüência), utilizando cada uma das quatro amostras distintas de 2.10^4bits, foram aceitos ao final da realização dos mesmos, sendo que todos foram efetuados utilizando-se um nível de significância α igual a 0.05.

5.2.4 Bateria de Testes FIPS PUB 140-2

A bateria de testes definida no padrão FIPS PUB 140-2 [17] especifica quatro testes para a serem aplicados sobre amostras produzidas pelo gerador de números avaliado, sendo que a falha em qualquer um dos testes definidos pelo padrão acarreta na falha do próprio gerador.

Os testes são executados a partir de amostras produzidas pelo gerador de comprimento 2.10^4bits, de modo que foram utilizadas as mesmas seqüências utilizadas no item 5.2.3, nomeadas de acordo com a Tabela 5.3.

5.2.4.1 Monobit

O teste *monobit* define que a quantidade de uns da amostra, denotada *X*, deve estar contida no intervalo:

$$9725 < X < 10275 \tag{51}$$

Os resultados obtidos nas amostras avaliadas são apresentados na Tabela 5.8.

Tabela 5.8: Resultados do teste *monobit* FIPS 140-2.

Nome da amostra	*X*
Escritório	9971
Lar doce Lar	9975
Litoral	10023
Casa de Campo	10118

Como todos os valores encontram-se dentro do intervalo esperado, pode-se afirmar que todas as amostras avaliadas passam no teste *monobit* definido pelo padrão FIPS 140-2.

5.2.4.2 Poker test

O *poker test* define que a seqüência de 2.10^4bits seja dividida em 5.10^3 seqüências consecutivas de comprimento 4bits, existindo assim 2^4 possíveis seqüências de bits, cada uma definida por *f(i)*, sendo *i* um valor contido no intervalo:

$$0 \le i \le 15 \tag{52}$$

O seguinte cálculo, dado na equação (53), é então executado para cada umas das amostras.

$$X = \frac{16}{5000} \cdot \left(\sum_{i=0}^{15} \left[f(i) \right]^2 \right) - 5000 \tag{53}$$

De modo que *X* deve estar contido no intervalo:

$$2{,}16 < X < 46{,}17 \tag{54}$$

Os resultados obtidos nas amostras avaliadas são apresentados na Tabela 5.9.

Tabela 5.9: Resultados do *poker test* FIPS 140-2.

Nome da amostra	*X*
Escritório	12,79
Lar doce Lar	11,75
Litoral	13,42
Casa de Campo	16,76

Como todos os valores encontram-se dentro do intervalo esperado, pode-se afirmar que todas as amostras avaliadas passam no *poker test* definido pelo padrão FIPS 140-2.

5.2.4.3 Runs test

O *runs test* verifica a taxa de incidência de blocos e das lacunas de diferentes tamanhos encontrados na seqüência de 2.10^4bits produzida pelo gerador, sendo que o comprimento dos blocos e das lacunas varia entre uma e seis unidades. Neste teste, blocos e lacunas de comprimento superior a seis unidades são considerados de comprimento seis.

A taxa de incidência dos blocos e lacunas deve estar contida nos intervalos definidos na Tabela 5.10, abaixo.

Tabela 5.10: Limites das taxas de incidência de blocos e lacunas.

Comprimento em bits	Taxa de Incidência
1	2315 – 2685
2	1114 – 1386
3	527 – 723
4	240 – 384
5	103 – 209
6+	103 – 209

Os resultados obtidos nas amostras avaliadas são apresentados na Tabela 5.11.

Tabela 5.11: Incidência de blocos e colunas nas amostras.

	Escritório		*Lar doce Lar*		*Litoral*		*Casa de Campo*	
	Blocos	Lacunas	Blocos	Lacunas	Blocos	Lacunas	Blocos	Lacunas
1	2501	2545	2521	2491	2469	2506	2478	2440
2	1294	1207	1251	1208	1258	1249	1211	1248
3	638	662	603	684	647	643	622	666
4	303	299	286	292	293	303	324	306
5	139	150	147	143	141	151	167	147
6+	150	162	178	168	188	144	161	156

A verificação da taxa de incidência de blocos e colunas de diferentes comprimentos nas quatro amostras analisadas, ilustrada na Tabela 5.11 e comparada com os valores definidos na Tabela 5.10, demonstram que a quantidade de blocos e lacunas para todas as amostras testadas encontram-se dentro dos intervalos esperados, podendo-se então afirmar que todas passam com sucesso no *runs test* definido pelo padrão FIPS 140-2.

5.2.4.4 Long run test

O *long run test* define que o tamanho do maior bloco ou da maior lacuna encontrados em uma amostra não deve exceder 26bits de comprimento.

O tamanho dos maiores blocos e lacunas encontrados para cada uma das amostras testadas é apresentado na Tabela 5.12.

Tabela 5.12: Tamanho dos maiores blocos e lacunas.

	Escritório	*Lar doce Lar*	*Litoral*	*Casa de Campo*
Maior bloco	19bits	17bits	14bits	13bits
Maior lacuna	11bits	16bits	13bits	15bits

Como pode ser verificado na Tabela 5.12, nenhuma das amostras apresentou blocos ou lacunas de comprimento superior a 26bits, de modo que todas passam no *long run test* definido pelo padrão FIPS 140-2.

5.2.4.5 Conclusões obtidas da bateria de testes FIPS 140-2

Todas as quatro amostras de 2.10^4bits de comprimento, as mesmas utilizadas no item 5.2.3, passaram nos testes da bateria FIPS 140-2.

Deve-se ressaltar que o padrão FIPS 140-2 recomenda que, para aplicações com fortes requisitos de segurança (nível de segurança 4), seja executada a bateria de testes cada vez que o dispositivo que hospedar o gerador seja ativado. Para outras aplicações com requisitos de segurança menos restritos (nível de segurança 3) a realização da bateria de testes deve ser feita sob demanda, ou seja, quando desejado pelo usuário[50].

5.2.5 Conclusões sobre os testes realizados sobre o PRNG

Amostras produzidas pelo gerador de números utilizado no mecanismo de Domínios Virtuais foram submetidas aos testes de aderência à distribuição χ^2, apresentado no item 5.2.2, a bateria FIPS 140-2, no item 5.2.4, e também a outros testes independentes, relacionados no item 5.2.3.

É importante ressaltar que o sucesso obtido por todas as amostras submetidas aos testes realizados é condição necessária para que o gerador utilizado possa ser considerado um PRNG, mas não é capaz de garantir que este gerador é realmente um PRNG [74] [75].

No entanto, a realização de múltiplos testes, utilizando diferentes amostras, e o sucesso obtido em todos, como apresentados no decorrer deste capítulo, nos itens 5.2.2, 5.2.3, e 5.2.4, apresentam evidências de que o gerador utilizado possa realmente ser considerado um PRNG.

5.3 Operações sobre Informações Armazenadas

Os testes relacionados às operações sobre informações armazenadas correspondem à verificação do correto funcionamento das classes

---

[50] O padrão FIPS 140-2 classifica os requisitos de segurança em quatro níveis distintos, no qual o nível 4 corresponde à maior exigência em relação à segurança, enquanto que o nível 1, a menor.

responsáveis pelo armazenamento de informações, ou seja, as classes *SeedTable* e *MessageIndex*.

Estas informações são as sementes utilizadas para identificar os diferentes domínios virtuais e também o índice das mensagens transmitidas.

Estes testes são realizados antes da integração destas classes com as demais que compõem o mecanismo de Domínios Virtuais, de modo a avaliá-las individualmente de acordo com seus requisitos.

5.3.1 Inclusão e Exclusão de Sementes

A inclusão e exclusão de sementes é executada através de métodos presentes no objeto instanciado da classe *SeedTable*. Esta classe foi testada através da utilização de uma simples interface homem-máquina, apresentada na Figura 5.13.

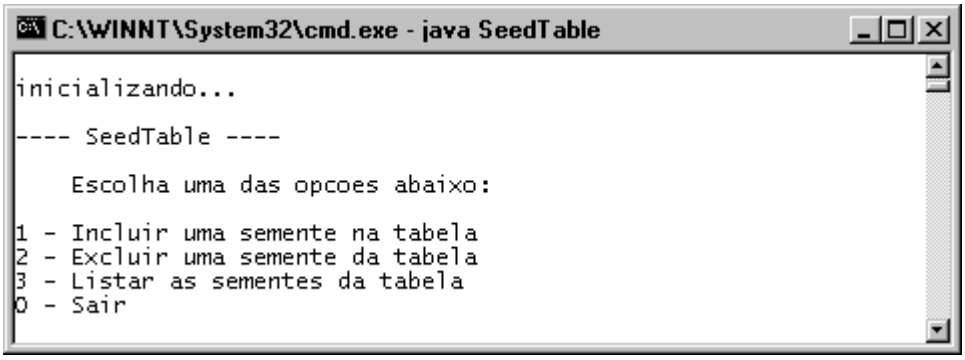

Figura 5.13: Menu da interface de um objeto da classe *SeedTable*.

5.3.1.1 Inclusão de sementes.

A inclusão das sementes é efetuada através da primeira opção do menu principal, apresentado na Figura 5.13. As sementes são identificadas por um nome e possuem um valor associado a ele, definido, no caso, pelo usuário. A Figura 5.14 apresenta a inclusão de uma semente denominada *Escritório*.

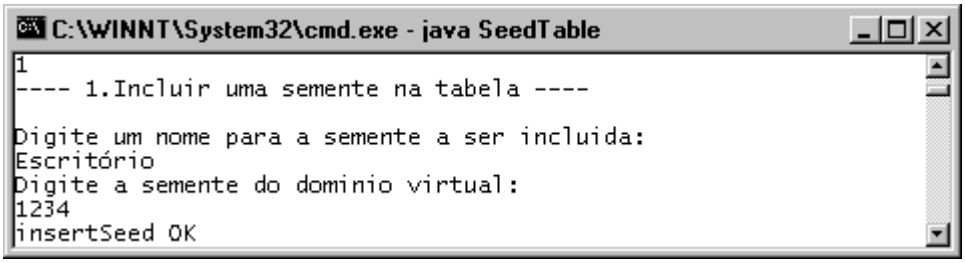

Figura 5.14: Inclusão de uma semente na tabela.

Outras sementes também são incluídas na tabela, e seu conteúdo, após a adição de todas as demais sementes, é apresentado na Figura 5.15.

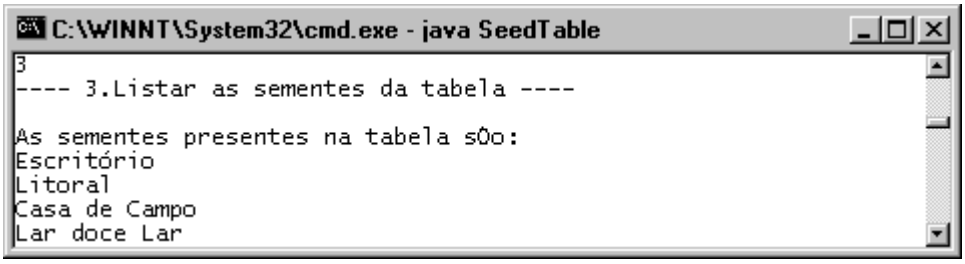

Figura 5.15: Sementes presentes na tabela.

5.3.1.2 Exclusão de uma semente.

A exclusão das sementes é efetuada através da segunda opção do menu principal, apresentado na Figura 5.13. A Figura 5.16 apresenta a exclusão de uma semente denominada *Litoral*.

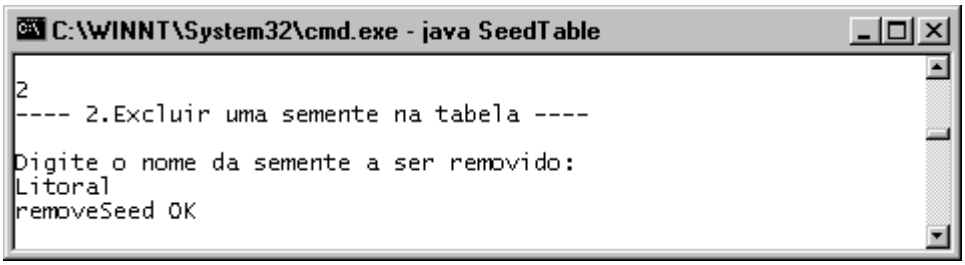

Figura 5.16: Exclusão de uma semente da tabela.

A tabela resultante, apresentada na Figura 5.17, confirma a operação de exclusão da semente.

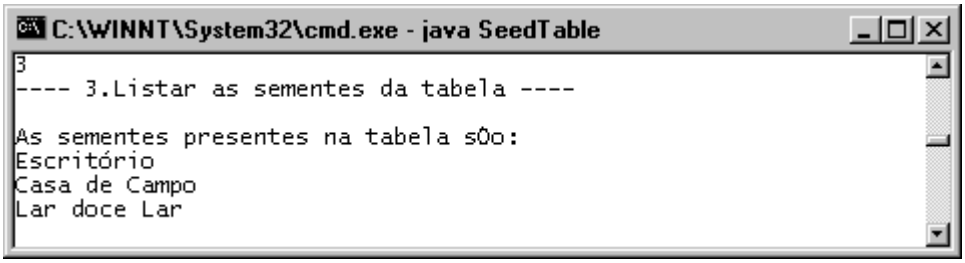

Figura 5.17: Sementes presentes na tabela após o processo de exclusão.

Após a exclusão de todas as sementes, a tabela resultante deve apresentar conteúdo nulo, como é confirmado na Figura 5.18.

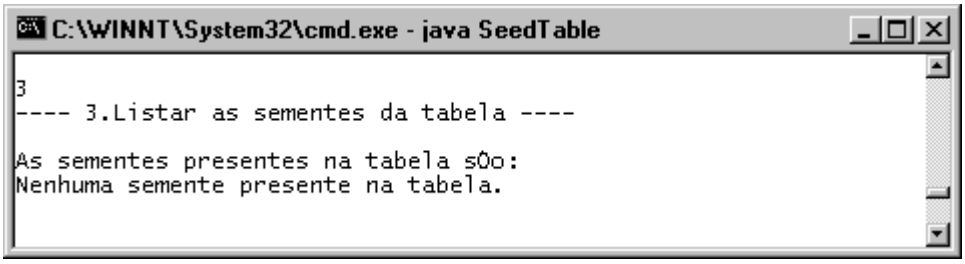

Figura 5.18: Conteúdo da tabela após a exclusão de todas as sementes.

5.3.2 Inserção e Remoção de Mensagens

A inclusão e a exclusão de mensagens é executada através de métodos presentes no objeto instanciado da classe *MessageIndex*. Os testes são feitos para cada uma das três tabelas utilizadas pelo objeto. A classe foi testada através da utilização de uma simples interface homem-máquina, apresentada na Figura 5.19.

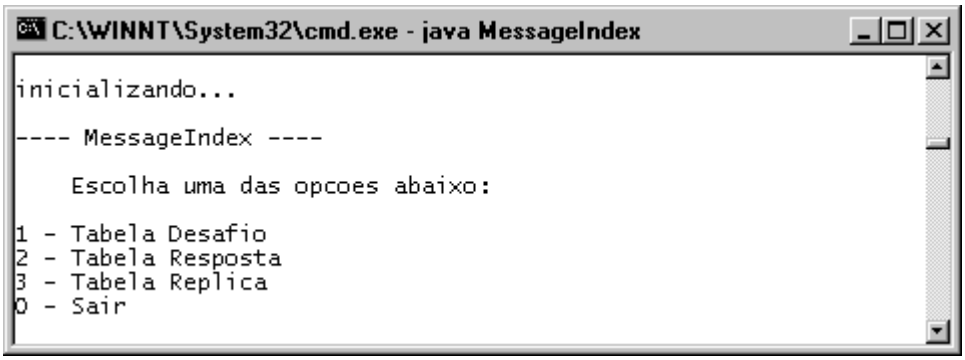

Figura 5.19: Menu da interface de um objeto da classe *MessageIndex*.

Como os métodos que manipulam as três tabelas que compõem um objeto da classe *MessageIndex* são idênticos, será apresentado apenas o teste

sobre operações em uma das tabelas. Sendo assim, uma tabela que manipula mensagens do tipo *desafio* será utilizada para a realização dos testes.

O acesso a tabela de mensagens do tipo *desafio* é feito através da escolha da primeira opção do menu principal, apresentado na Figura 5.19. As operações de manipulação do conteúdo da tabela são apresentadas na Figura 5.20.

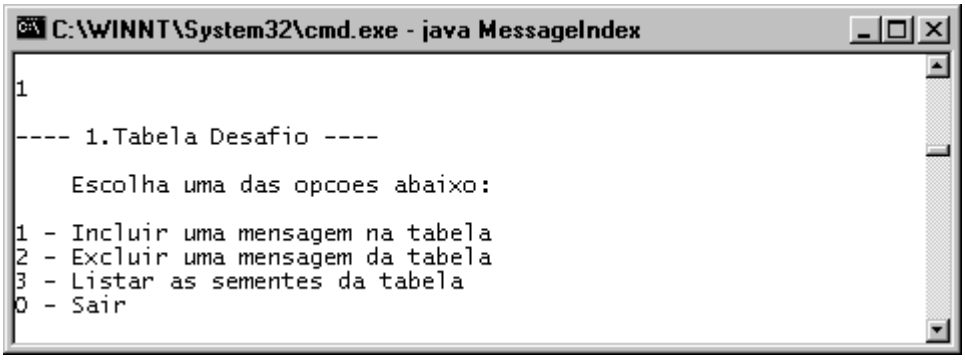

Figura 5.20: Operações sobre mensagens da Tabela Desafio.

5.3.2.1 Inclusão de uma mensagem

A inclusão de uma mensagem é efetuada através da primeira opção do menu apresentado na Figura 5.20. As mensagens são identificadas por um índice definido por valor numérico, escolhido, neste caso, pelo usuário. A Figura 5.21 apresenta a inclusão de uma mensagem identificada pelo valor *1234*.

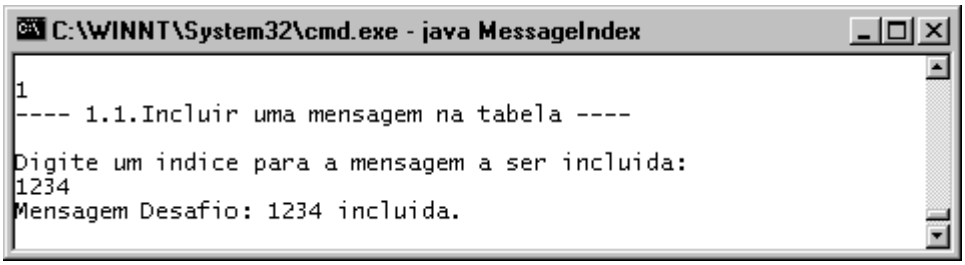

Figura 5.21: Inclusão de uma mensagem na tabela.

Outras mensagens são incluídas na mesma tabela, e seu conteúdo, após a adição de todas as demais mensagens, é apresentado na Figura 5.22.

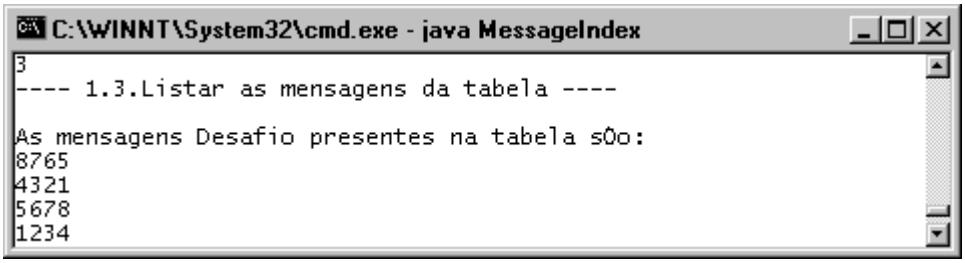

Figura 5.22: Mensagens presentes na tabela após o processo de inclusão.

5.3.2.2 Exclusão de uma mensagem

A exclusão de uma mensagem é efetuada através da segunda opção do menu apresentado na Figura 5.20. A Figura 5.23 apresenta a exclusão de uma mensagem identificada pelo valor *1234*.

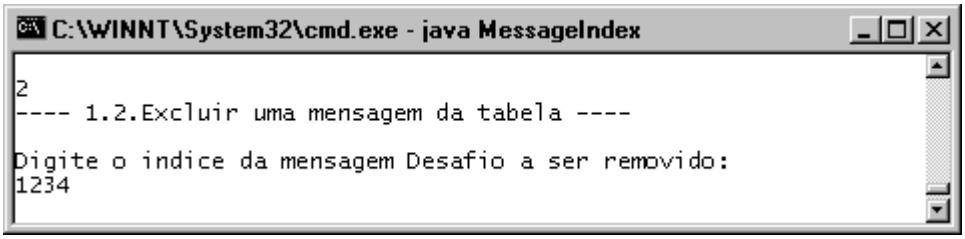

Figura 5.23: Exclusão de uma mensagem da tabela.

A tabela resultante, apresentada na Figura 5.24, confirma a operação de exclusão da mensagem.

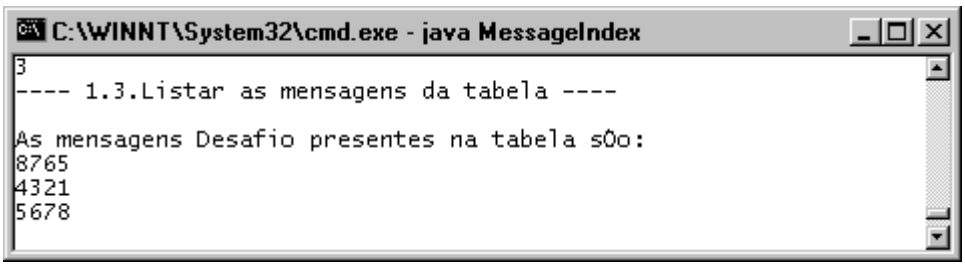

Figura 5.24: Mensagens presentes na tabela após a operação de exclusão.

Após a exclusão de todas as mensagens, a tabela resultante deve apresentar conteúdo nulo, como é confirmado na Figura 5.25.

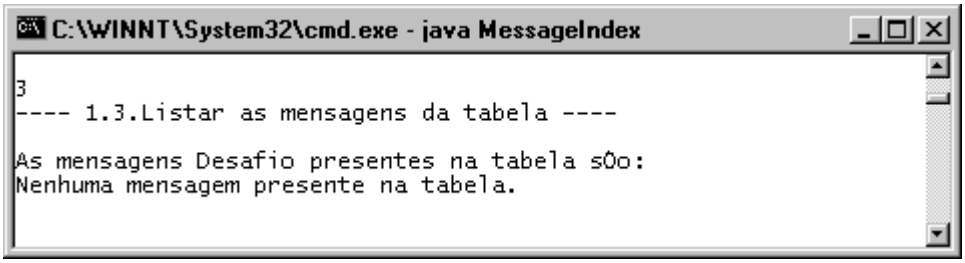

Figura 5.25: Conteúdo da tabela após a exclusão de todas as mensagens.

5.4 Mecanismo de Domínios Virtuais

A avaliação final do protótipo do mecanismo de Domínios Virtuais pode ser realizada após a integração das diversas classes que o compõem.

O objetivo dos testes a serem realizados no decorrer deste item é verificar a correta operação das funcionalidades do mecanismo proposto, como a adição de novos domínios virtuais ou a alteração de seus parâmetros, por exemplo.

Uma interface homem-máquina simples foi adicionada ao mecanismo, como já citado no item 4.1.3, de forma a permitir a execução dos testes. A ordem da realização dos testes segue, de modo geral, a seqüência na qual os itens são apresentados no menu principal da interface, apresentado na Figura 5.26.

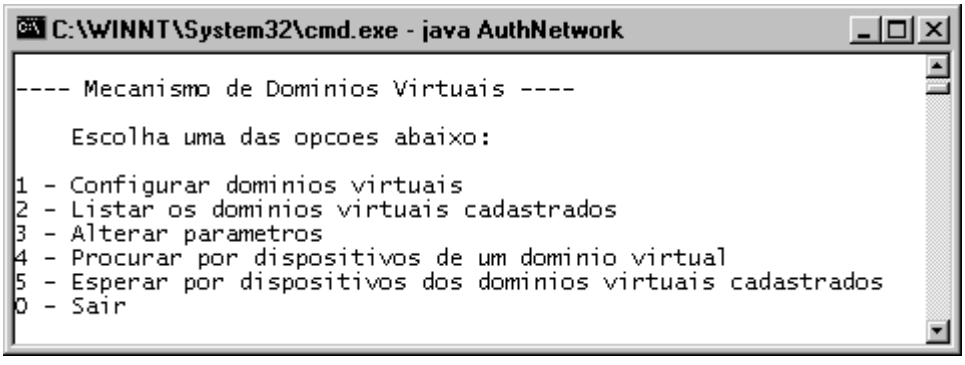

Figura 5.26: Menu da interface do mecanismo de Domínios Virtuais.

Deste modo, os primeiros testes a serem realizados são relativos à configuração dos domínios virtuais, ou seja, operações como a listagem, inclusão e exclusão. A alteração de parâmetros é testada a seguir, com a mudança da porta TCP utilizada na comunicação e o redimensionamento

da janela de tempo. Nota-se que todos os testes até este ponto podem ser realizados através da utilização de apenas um dispositivo.

Finalmente, o processo de identificação é testado através da utilização de dois dispositivos móveis se comunicando através de um enlace de comunicação sem fio IEEE 802.11b.

5.4.1 Configurando Domínios Virtuais

A adição e a remoção de domínios virtuais reconhecidos por um determinado dispositivo são efetuadas por determinação do usuário.

De acordo com os casos de uso definidos no item 4.4, e seus respectivos diagramas de seqüência, a inclusão e a exclusão de domínios virtuais será testada. Para a verificação do sucesso dos testes realizados, foi adicionado um método à classe *AuthNetwork* que permite a listagem dos domínios virtuais cadastrados.

A configuração do mecanismo de Domínios Virtuais corresponde à primeira opção do menu principal da interface da aplicação, apresentado na Figura 5.26. A Figura 5.27 apresenta o menu desta opção, que é o ponto de partida para as operações de adição e remoção de domínios virtuais.

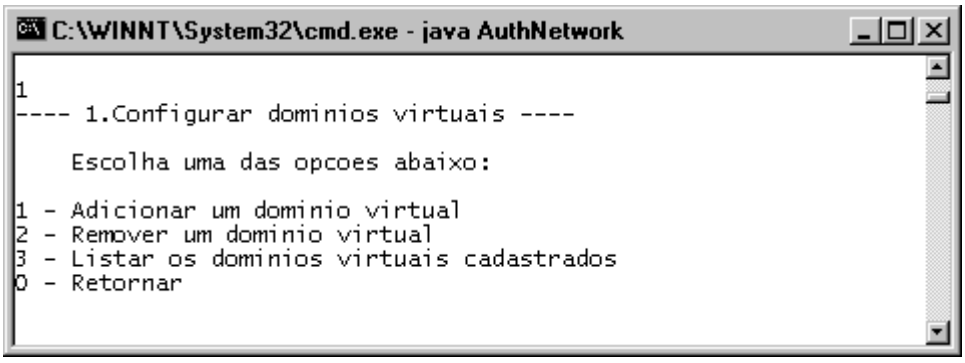

Figura 5.27: Opções de configuração dos domínios virtuais.

As opções utilizadas para a inclusão de um novo domínio virtual serão testadas para então ser verificado o funcionamento da exclusão de um domínio virtual existente.

5.4.1.1 Inclusão automática de um domínio virtual

A inclusão automática de um novo domínio virtual, ou seja, sem a definição explícita de uma semente a ser utilizada pelo gerador de números pseudo-aleatórios, necessita apenas da definição de seu nome. A semente a ser utilizada pelo gerador é produzida pelo próprio mecanismo, e possui 160bits de comprimento.

A Figura 5.28 apresenta a inclusão de um novo domínio virtual denominado *Escritório*, assim como a resposta do sistema, confirmando a operação.

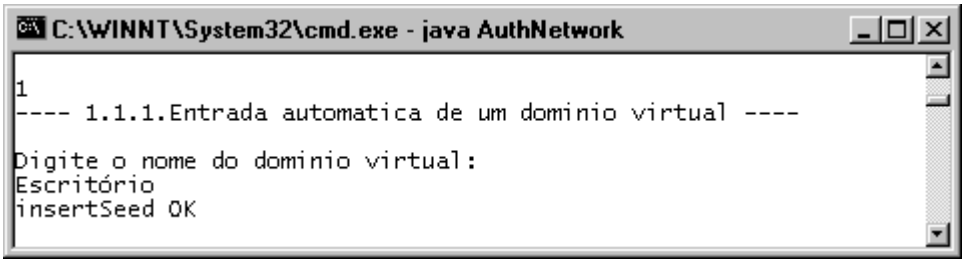

Figura 5.28: Inclusão automática de um domínio virtual.

5.4.1.2 Inclusão de um domínio virtual

A inclusão de um novo domínio virtual necessita da definição de um nome e de um valor para a semente a ser utilizada pelo PRNG. A utilização deste modo de inclusão permite, portanto, que o usuário possa definir a seqüência a ser utilizada pelo domínio virtual criado.

A Figura 5.29 apresenta a inclusão de um novo domínio virtual denominado *Lar doce Lar*, utilizando uma semente de valor *1234*.

A resposta do sistema, confirmando a operação, é apresentada na linha seguinte.

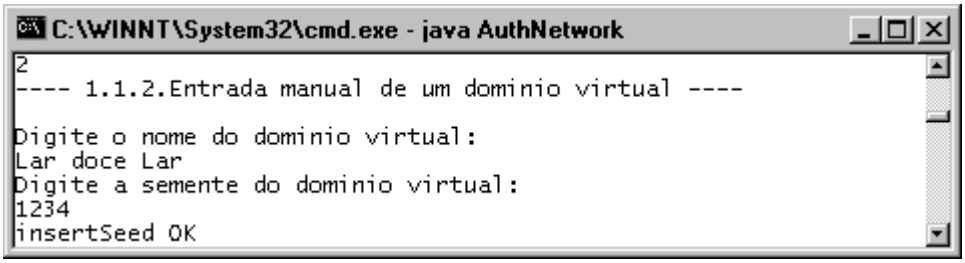

Figura 5.29: Inclusão de um domínio virtual.

5.4.1.3 Exclusão de um domínio virtual

Após a inserção de domínios virtuais, é possível removê-los do sistema.

Deste modo, após a inclusão dos domínios virtuais apresentados nos itens 5.4.1.1 e 5.4.1.2, outros dois domínios, denominados *Litoral* e *Casa de Campo*, são acrescentados. A Figura 5.30 apresenta os domínios virtuais cadastrados.

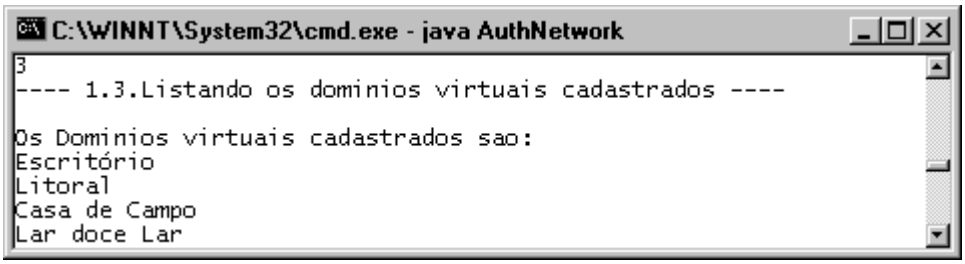

Figura 5.30: Domínios virtuais cadastrados.

A Figura 5.31 apresenta a exclusão do domínio virtual denominado *Litoral*. A resposta do sistema, confirmando a operação, é apresentada em seguida.

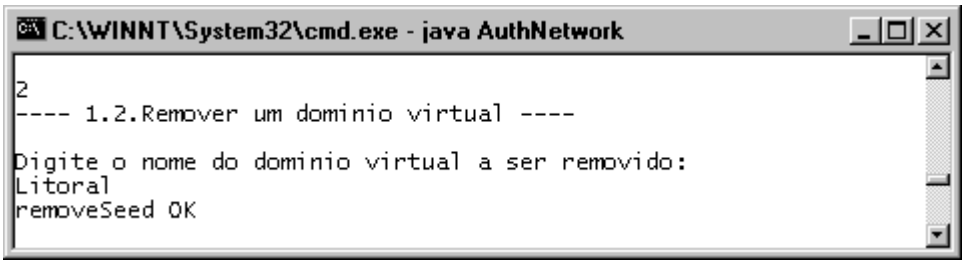

Figura 5.31: Remoção de um domínio virtual.

A Figura 5.32 apresenta os domínios virtuais cadastrados após a remoção do domínio virtual *Litoral*.

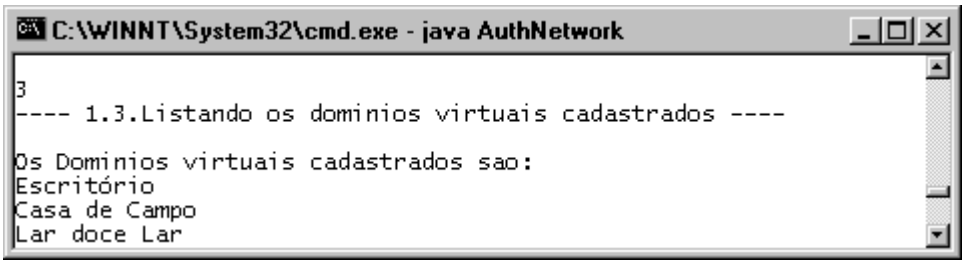

Figura 5.32: Domínios virtuais cadastrados após a remoção.

5.4.2 Alteração de Parâmetros

A alteração dos parâmetros de configuração do mecanismo, ou seja, a mudança da porta TCP utilizada na comunicação e a redefinição da janela de tempo, é efetuada seguindo opções do usuário. De acordo com os casos de uso definidos no item 4.4, e seus respectivos diagramas de seqüência, a alteração de parâmetros do mecanismo será testada.

A terceira opção do menu principal da interface da aplicação, apresentado na Figura 5.26, corresponde à alternativa para a alteração dos parâmetros de configuração do mecanismo de Domínios Virtuais. A Figura 5.33 apresenta o menu desta opção, que é o ponto de partida para suas operações.

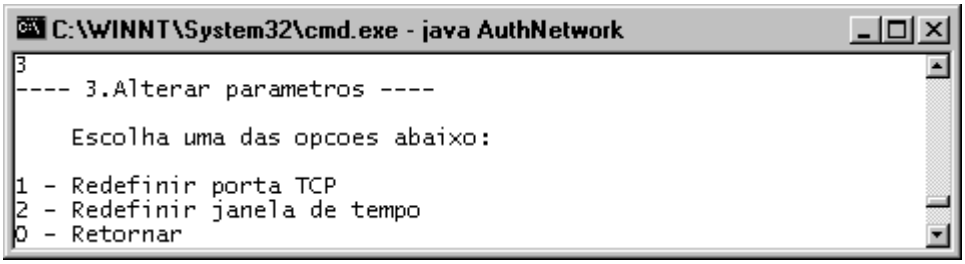

Figura 5.33: Menu principal da opção de alteração de parâmetros.

Os testes seguem a ordem das operações presentes no menu apresentado na Figura 5.33.

5.4.2.1 Porta TCP

A redefinição da porta de comunicação TCP utilizada pelo mecanismo proposto é apresentada na Figura 5.34. A resposta do sistema, confirmando a operação, segue na linha seguinte, indicando a nova porta de comunicação selecionada.

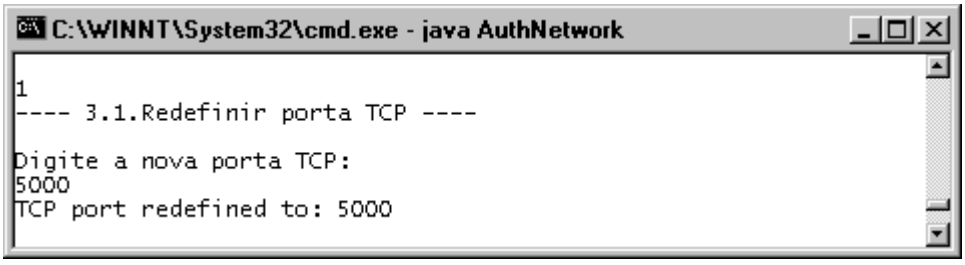

Figura 5.34: Redefinição da porta TCP.

5.4.2.2 Janela de tempo

A redefinição do tamanho da janela de tempo utilizada pelo mecanismo é apresentada na Figura 5.35. O tamanho da janela de tempo é dado em milésimos de segundo, dando flexibilidade ao usuário.

A resposta do sistema, confirmando a operação, segue na linha seguinte, indicando o novo tamanho definido para a janela de tempo a ser utilizada pelo mecanismo.

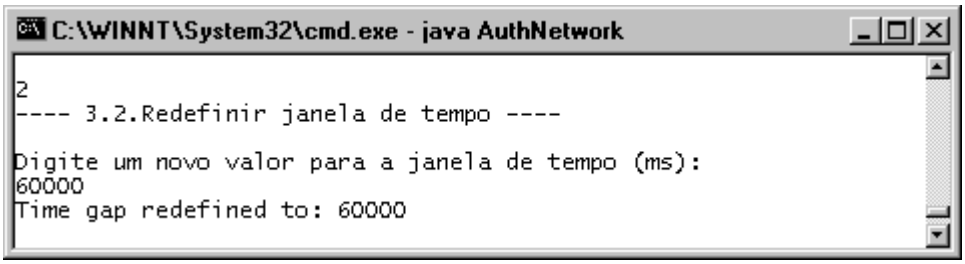

Figura 5.35: Redefinição da janela de tempo.

5.4.3 Reconhecimento de um Domínio Virtual

Após a definição dos parâmetros do mecanismo de Domínios Virtuais, é possível iniciar a localização por outros dispositivos pertencentes a um domínio virtual cadastrado.

A realização deste teste foi efetuada através da utilização de dois dispositivos móveis que possuem alguns domínios virtuais em comum. A Figura 5.36 apresenta estes dispositivos, assim como seus domínios virtuais cadastrados.

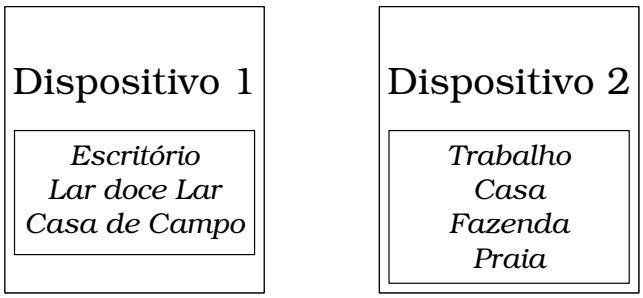

Figura 5.36: Dispositivos e seus domínios virtuais.

Os domínios virtuais *Escritório*, do primeiro dispositivo, e *Trabalho*, do segundo dispositivo, correspondem, na verdade, ao mesmo domínio virtual.

É importante lembrar que o nome associado a um determinado domínio virtual trata-se de uma escolha atribuída pelo usuário, ou seja, não é uma imposição do mecanismo, e que um domínio virtual é, na verdade, definido pela seqüência de valores gerados, que é, por sua vez, determinada pela semente do PRNG.

Assim como as denominações *Escritório* e *Trabalho* referem-se a um mesmo domínio virtual, *Lar doce Lar* e *Casa* também correspondem a um mesmo domínio virtual, valendo a mesma relação para *Casa de Campo* e *Fazenda*. O domínio virtual *Praia* é o único que existe em apenas um dispositivo. A Figura 5.37 apresenta a relação dos domínios virtuais cadastrados no primeiro dispositivo.

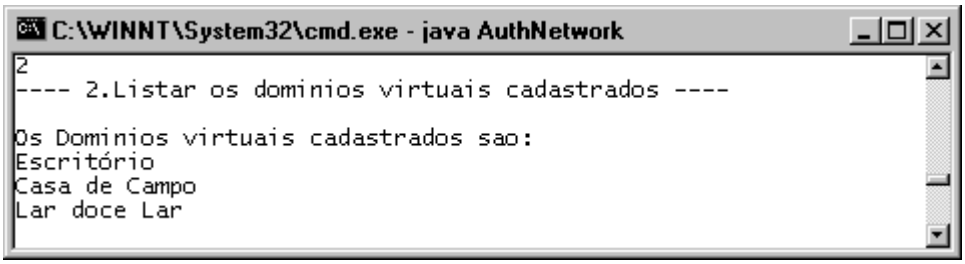

Figura 5.37: Domínios virtuais cadastrados no primeiro dispositivo.

A seguir, a Figura 5.38 apresenta a relação dos domínios virtuais cadastrados no segundo dispositivo.

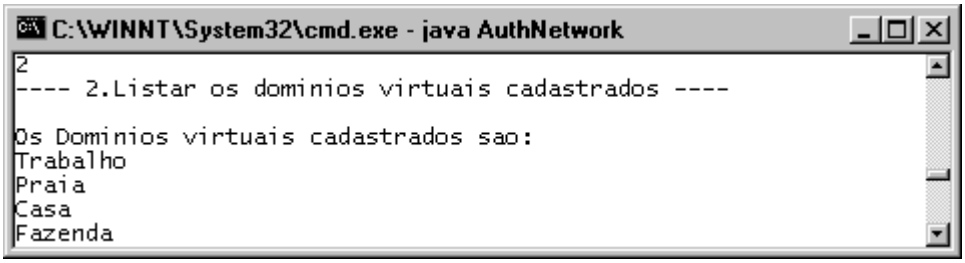

Figura 5.38: Domínios virtuais cadastrados no segundo dispositivo.

O relógio dos dispositivos apresenta uma defasagem de apenas alguns segundos, de modo que ambos estejam dentro da janela de tempo definida no item 5.4.2.2, de um minuto.

Segundo a máquina de estados do mecanismo de Domínios Virtuais, apresentada na Figura 4.1, o estado seguinte ao estado inicial pode ser

definido pelo recebimento de uma mensagem do tipo *desafio* de um segundo dispositivo, ou pelo envio de uma mensagem deste mesmo tipo.

Deste modo, após a escolha da quinta opção do menu principal do mecanismo, referente à Figura 5.26, o dispositivo passa a esperar conexões na porta de comunicação TCP previamente definida no item 5.4.2.1. A Figura 5.39 indica que o dispositivo passou a esperar mensagens oriundas de dispositivos virtuais cadastrados. A espera por dispositivos pertencentes a um domínio virtual cadastrado é executada em ambos os dispositivos.

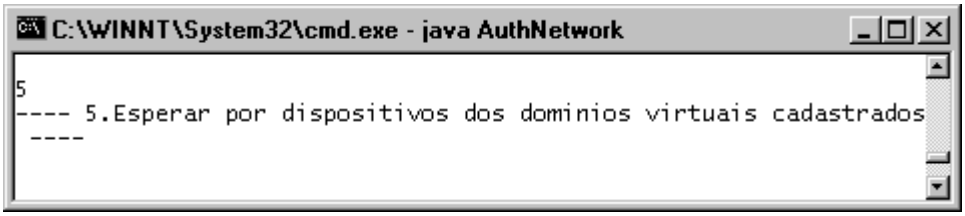

Figura 5.39: Dispositivos passam a esperar mensagens.

No entanto, apesar de ambos os dispositivos estarem esperando conexões, isto não os impede enviar requisições, ou seja, mensagens do tipo *desafio*, com o objetivo de procurar por outros dispositivos pertencentes a um domínio virtual cadastrado. É importante lembrar que o mecanismo de domínios virtuais é uma aplicação do tipo *peer-to-peer*.

Assim, através da escolha, em apenas um dos dispositivos, da quarta opção do menu principal do mecanismo, referente à Figura 5.26, a verificação tem início. A Figura 5.40 apresenta este processo sendo iniciado pelo segundo dispositivo, que irá verificar se o primeiro, denominado *edelweiss*, pertence ao domínio virtual *Trabalho*.

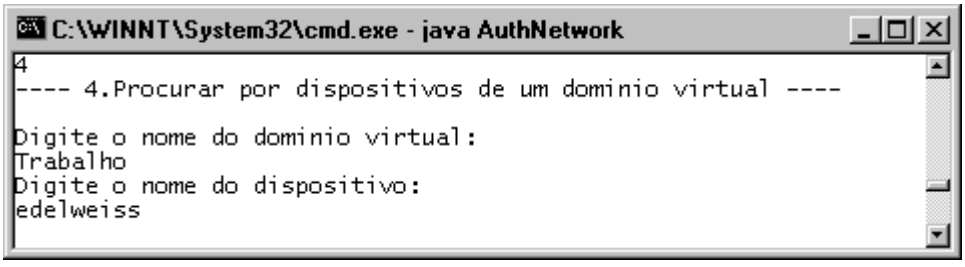

Figura 5.40: Procura por dispositivo pertencente ao domínio *Trabalho*.

É importante lembrar que o reconhecimento entre dispositivos é sempre mútuo, ou seja, ele sempre ocorre em ambos os sentidos. Sendo assim, não

existe reconhecimento unilateral de um dispositivo, pois ambos devem ser identificados como participantes de um mesmo domínio virtual.

A Figura 5.41 apresenta as mensagens geradas pelo primeiro dispositivo, denominado *edelweiss*, durante o reconhecimento do segundo.

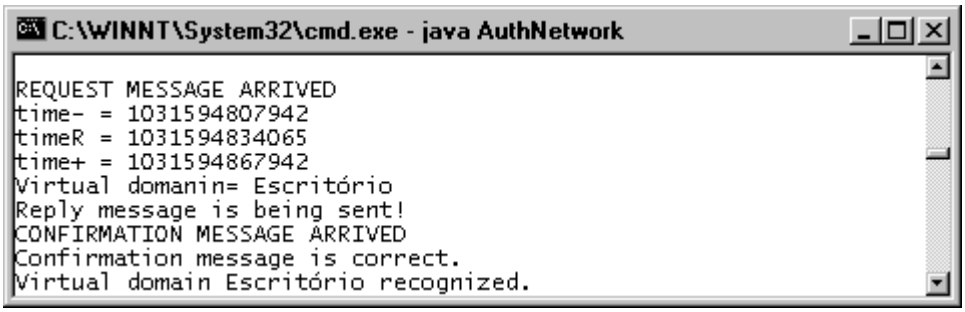

Figura 5.41: Reconhecimento do dispositivo requisitante.

A primeira linha da Figura 5.41 indica o recebimento de uma mensagem do tipo *desafio*[51], enquanto as três linhas subseqüentes representam os limites da janela de tempo e o tempo associado à mensagem recebida. A quinta linha indica que, a princípio, foi reconhecido que a mensagem recebida do tipo *desafio* é, provavelmente, oriunda de um dispositivo pertencente ao domínio virtual *Escritório*.

A sexta linha da Figura 5.41 indica que uma mensagem do tipo *resposta*[52] foi enviada, enquanto que a sétima linha aponta o recebimento de uma mensagem do tipo *réplica*[53] e a oitava linha indica que os dados nela recebidos são realmente válidos. A última linha da Figura 5.41 confirma e sinaliza o reconhecimento de um dispositivo pertencente ao domínio virtual *Escritório*.

Nota-se que a diferença entre o centro do intervalo definido pela janela de tempo, calculado pela média aritmética entre os seus limites, e o valor de tempo recebido é menor que quatro segundos, confirmando a pequena

---

[51] O termo *request message* que aparece na Figura 5.41 deve ser entendido como um sinônimo para a mensagem do tipo *desafio*. Na mesma figura, esta relação existe também para o termo *reply message* que está associado às mensagens do tipo *resposta*, e *confirmation message* que denomina as mensagens do tipo *réplica*.
[52] Ver nota de rodapé 51.
[53] Ver nota de rodapé 51.

defasagem entre os relógios dos dispositivos testados, como apresentado anteriormente.

É possível notar também que a janela de tempo é, como definido, de um minuto, ou 6.10^4 milésimos de segundo.

A seguir, a Figura 5.42 apresenta as mensagens geradas pelo segundo dispositivo, durante o processo de reconhecimento do primeiro.

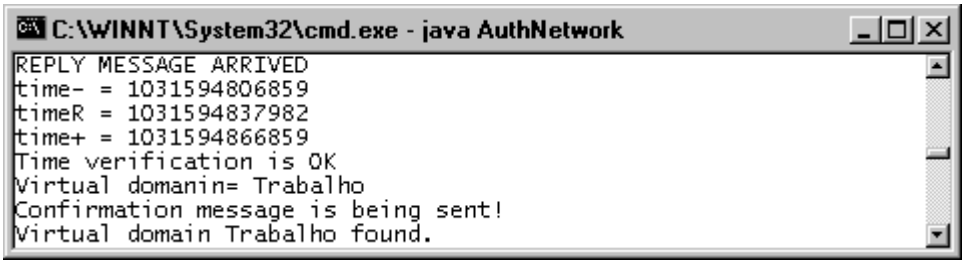

Figura 5.42: Reconhecimento de um dispositivo.

A primeira linha da Figura 5.42 indica o recebimento de uma mensagem do tipo *resposta*, enquanto as quatro linhas subseqüentes apresentam informações relativas aos limites da janela de tempo. A sexta linha indica que, a princípio, foi reconhecido que a mensagem recebida do tipo *resposta* é, provavelmente, oriunda de um dispositivo pertencente ao domínio virtual *Trabalho*.

A sétima linha da Figura 5.42 indica que uma mensagem do tipo *réplica* foi enviada, enquanto que a última linha aponta que o dispositivo foi reconhecido como pertencente ao domínio virtual *Trabalho*.

5.4.3.1 As janelas de tempo

Para verificar o funcionamento das janelas de tempo do mecanismo, o relógio do sistema do segundo dispositivo foi atrasado em cerca de quatro minutos, e a verificação foi iniciada novamente.

A Figura 5.43 indica que o índice de tempo recebido na mensagem do tipo *desafio* enviada pelo segundo dispositivo encontra-se fora dos limites definidos para a janela de tempo fixada pelo primeiro dispositivo. Conseqüentemente, a conexão TCP foi finalizada.

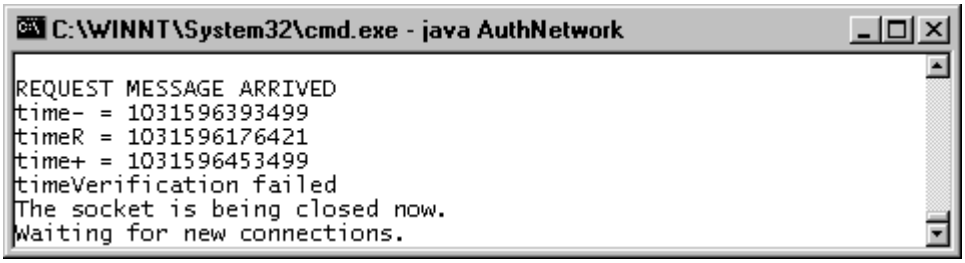

Figura 5.43: Mensagem recebida fora dos limites da janela de tempo.

É possível verificar que o tamanho da janela de tempo, definido pela diferença entre seus limites superior e inferior, permanece igual a um minuto. Nota-se também que a diferença entre o centro da janela de tempo definido pela média aritmética de seus limites e o índice de tempo recebido na mensagem *desafio*, oriunda do segundo dispositivo, é de cerca 25.10^4 milésimos de segundo, ou seja, pouco mais de quatro minutos, como era esperado.

5.4.3.2 Não reconhecimento de um domínio virtual

Um domínio virtual pode não ser reconhecido, caso este não esteja cadastrado.

Sendo assim, caso o segundo dispositivo procure pelo domínio virtual *Praia*, como apresentado na Figura 5.44, no primeiro dispositivo, não deve ocorrer o reconhecimento do domínio virtual em questão.

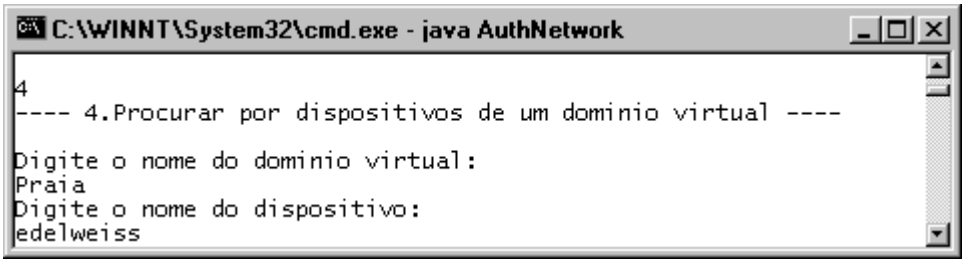

Figura 5.44: Procurando pelo domínio *Praia*.

É importante lembrar que o domínio virtual equivalente ao domínio *Praia*, o domínio *Litoral*, foi retirado da lista dos domínios virtuais cadastrados, como apresentado no item 5.4.1.3. O não reconhecimento do domínio virtual, pelo segundo dispositivo, é apresentado na Figura 5.45.

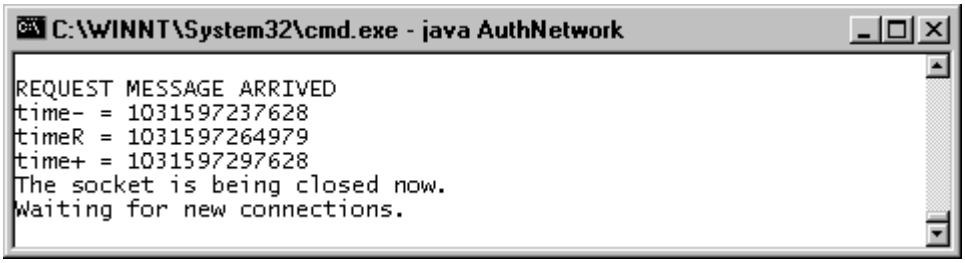

Figura 5.45: Domínio virtual não reconhecido.

A primeira linha da Figura 5.45 indica o recebimento de uma mensagem do tipo *desafio*, enquanto que as três linhas seguintes mostram que o índice de tempo da mensagem recebida encontra-se dentro da janela de tempo. Como os dados recebidos não correspondem aos valores esperados para nenhum dos domínios virtuais reconhecidos, a mensagem é descartada e a conexão TCP finalizada.

5.5 Considerações sobre os Testes e Resultados

No decorrer deste capítulo procurou-se testar os diversos blocos que compõem a implementação do mecanismo de Domínios Virtuais.

Inicialmente, foi testado o PRNG selecionado no item 3.3.1, uma variação do PRNG DSA, de modo a verificar se a alteração executada sobre o algoritmo comprometeu, de algum modo, as características pseudo-randômicas do gerador.

Todas as amostras produzidas a partir do gerador de números passaram com sucesso em todos os testes realizados. É importante ressaltar que a os resultados obtidos são condição necessária para que o gerador utilizado possa ser considerado um PRNG, mas não é capaz de garantir que este é, de fato, um PRNG. Este fato, no entanto, não desqualifica o gerador utilizado, já que não é possível obter-se uma prova definitiva sobre qualquer outro gerador existente [74] [75].

Os testes realizados sobre as operações em informações armazenadas foram realizados através da manipulação de dados em objetos instanciados das classes a serem avaliadas. Os resultados obtidos nos testes sobre estas

classes, *SeedTable* e *MessageIndex*, comprovaram o correto funcionamento das mesmas.

Os últimos testes foram realizados sobre a aplicação propriamente dita, ou seja, sobre a integração das diversas classes que compõem o protótipo do mecanismo de Domínios Virtuais. Estes testes incluíram todos os aspectos nele implementados, como a configuração dos parâmetros de funcionamento do mecanismo, a inclusão e a remoção de domínios virtuais e a verificação da correta operação das janelas de tempo utilizadas. Todos os testes foram bem sucedidos.

Deve-se destacar que, dentre os testes realizados, os resultados de maior relevância referem-se aos obtidos nos testes do PRNG, pois, como já afirmado anteriormente no item 5.2, a existência do mecanismo de Domínios Virtuais é fundamentada em um gerador de números pseudo-aleatórios, que possua propriedades criptográficas e que seja capaz de aceitar um parâmetro de entrada arbitrário e de conhecimento público.

Portanto, apesar de nenhum dos testes realizados poder fornecer um resultado definitivo sobre o gerador utilizado, pode-se afirmar também que, como todos os testes realizados até este momento foram bem sucedidos e, enquanto não forem obtidos resultados que provem o contrário, este se pode ser tratado como um gerador adequado para o mecanismo.

Capítulo 6

Considerações Finais

"Things do not change; we change."

Henry David Thoreau

O mecanismo de Domínios Virtuais proposto nesta dissertação de mestrado é submetido a uma última e conclusiva avaliação frente aos objetivos propostos para o mesmo, aos requisitos de segurança desejados e aos demais desafios impostos pelas características presentes em uma rede móvel ad hoc.

Neste capítulo final, é ainda apresentada uma visão geral sobre as contribuições acadêmicas fornecidas pelo mecanismo de Domínios Virtuais através de uma rápida descrição do posicionamento do mecanismo junto aos demais trabalhos propostos.

Propostas sobre a continuidade do trabalho antecedem o último item desta dissertação, que apresenta uma conclusão geral sobre o trabalho desenvolvido.

6.1 Discussão sobre os Resultados Obtidos

O mecanismo de Domínios Virtuais cumpre os objetivos descritos no item 1.1 ao criar condições para que informações relativas à um domínio virtual, como os dispositivos que a compõem ou os serviços nela disponíveis, permaneçam restritos aos seus participantes e, ainda, ao permitir que estes possam se movimentar sem que seja possível seu rastreamento, ou seja, anonimamente.

As questões de segurança em redes móveis ad hoc, apresentadas no item 2.2, também são respeitadas pelo mecanismo de Domínios Virtuais, já que

a divisão e a junção de um domínio virtual ocorre de modo natural, de modo a não existir a necessidade de definições de fronteiras.

A quantidade de informações que devem ser configuradas é mínima, correspondendo, no máximo, à definição de uma semente a ser utilizada pelo PRNG, que pode ser vista como um simples segredo a ser digitado, tornando a tarefa intuitiva e natural.

Outro ponto relevante é a completa e total ausência de elementos centrais no mecanismo de Domínios Virtuais, sendo cada dispositivo responsável pela sua segurança.

Deve-se notar que todos os desafios impostos ao desenvolvimento de um mecanismo de segurança para redes móveis ad hoc são superados sem que nenhuma funcionalidade ou característica das redes móveis ad hoc fosse tolhida ou limitada.

O protótipo do mecanismo de Domínios Virtuais também atendeu ao seu propósito, definido no item 1.1, verificando a viabilidade da implementação do mecanismo em questão. É importante lembrar que a implementação foi desenvolvida como um módulo de segurança a ser adicionado em uma completa arquitetura de segurança [15].

6.2 Contribuições e Posicionamento do Trabalho

Esta dissertação posiciona-se entre os demais trabalhos relacionados à segurança em redes móveis ad hoc de modo singular, criando uma barreira inicial a ataques externos sem que exista o comprometimento de quaisquer características que definem uma rede ad hoc, como a sua independência de entidades centrais, e ainda criando condições para que seja possível a proteção das informações relativas à movimentação de seus dispositivos e, portanto, possibilitando a existência do anonimato.

Os chamados domínios de rede virtuais, ou seja, dispositivos que compartilham um mesmo segredo, sendo assim capazes de gerar as mesmas seqüências no decorrer do tempo, sem a necessidade de uma perfeita sincronização de seus relógios, podem oferecer proteção a ataques externos através da utilização destas seqüências de números.

A implementação executada nesta dissertação utiliza, como segredo, valores pseudo-randômicos de 120bits de comprimento, retirados de uma seqüência de 160bits produzida através da utilização de uma variação do DSA PRNG, atualizada a cada milésimo de segundo.

A seqüência de números produzida pelo mecanismo de Domínios Virtuais é definida pela semente do PRNG utilizado, de modo que, quanto maior e mais forte for a semente, menor é a probabilidade de sua descoberta através de um ataque de força bruta. Deste modo, o mecanismo foi preparado para, além de poder aceitar uma semente definida pelo usuário, produzir sementes longas e de conteúdo aparentemente randômico.

A geração de sementes pelo mecanismo proposto necessita, no entanto, de um método externo que a complemente, possibilitando a distribuição das sementes produzidas, como a proposta de autenticação inicial do modelo de segurança *Resurrecting Duckling* apresentada no item 2.3.1, ou algum dos mecanismos de distribuição de chaves apresentados no item 2.3.5.

É importante ressaltar que o mecanismo de Domínios Virtuais não permite a exata identificação de um determinado dispositivo, restringindo-se apenas a identificá-lo como participante ou não de um determinado domínio virtual. Para que esta identificação seja possível, é necessária a utilização de certificados digitais e, portanto, uma inevitável entidade certificadora, que pode, adequando-se às redes móveis ad hoc, ser distribuída, como apresentado na proposta de Zhou e Haas [14], ou estar distribuída, como no Modelo de Segurança para Redes Móveis Ad Hoc [15] [16], do qual este mecanismo é parte componente.

Assim sendo, é possível verificar que mecanismo de Domínios Virtuais posiciona-se à frente dos demais, oferecendo uma primeira proteção aos dispositivos pertencentes a um domínio virtual, sendo fortalecido por outros mecanismos que permitem a distribuição de sementes, por exemplo, e complementado por outros, através da utilização de certificados digitais.

6.3 Continuidade

A próxima etapa prevista para este trabalho, imediatamente após sua conclusão, é a sua integração ao Modelo de Segurança para Redes Móveis Ad Hoc [15] [16].

Outra proposta, a ser executada futuramente, é o incremento do protótipo apresentado, aproximando-o cada vez mais da arquitetura proposta no item 3.3, que permita a manipulação dos campos de endereço físico da mensagem, de modo que o anonimato e a ocultação do usuário possam ser garantidos.

A quantificação da energia extra requerida na aplicação do mecanismo de Domínios Virtuais em um dispositivo corresponde a um trabalho futuro a ser executado, procurando garantir a viabilidade do mecanismo proposto em dispositivos móveis mais limitados em termos de consumo de energia.

A execução de uma maior quantidade de testes sobre amostras produzidas pelo PRNG utilizado no mecanismo de Domínios Virtuais é outra atividade a ser realizada futuramente, de modo a se obter mais resultados que confirmem que este pode realmente ser considerado um bom PRNG.

Lista de Referências

[1] PERKINS, C.E. Ad hoc networking – an introduction. In: PERKINS, C.E. **Ad hoc networking**. Upper Saddle River, New Jersey: Addison Wesley, 2000. p.1-28.

[2] HUBAUX, J.P., BUTTYAN, L., CAPKUN, S. The quest of security in mobile ad hoc networks. In: ACM SYMPOSIUM ON MOBILE AD HOC NETWORKING AND COMPUTING - MOBIHOC'01, Long Beach, Oct. 2001. **Proceedings**. Long Beach, California, USA, 2001.

[3] WEISER, M.; GOLD, R.; BROWN, J.S. The origins of the ubiquitous computing at PARC in the late 1980s. **IBM Systems Journal**, v.38, n.4, p.693-696, 1999.

[4] ARK, W.S.; and SELKER, T. A look at human interaction with pervasive computers. **IBM Systems Journal**, v.38, n.4, p.504-507, 1999.

[5] MILLER, B.A. Bluetooth application in pervasive computing. **An IBM Pervasive Computing White Paper**, Feb. 2000. Disponível em: <http://www-3.ibm.com/pvc/tech/bluetoothpvc.shtml>. Acesso em: 1º Set. 2001.

[6] IEEE COMPUTER SOCIETY. LAN/MAN Standards Committee. **ANSI/IEEE 802.11 Std**. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. New York, New York, USA: IEEE, Sep. 1999. 512p.

[7] IEEE COMPUTER SOCIETY. LAN/MAN Standards Committee. **ANSI/IEEE 802.11b Std**. Supplement to ANSI/IEEE Std. 802.11. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band. New York, New York, USA: IEEE, Sep. 1999. 89p.

[8] BLUETOOTH SIG. **Specification of the Bluetooth System**: wireless connections made easy. Core, version 1.1. Feb. 2001. 1082p. 2v.

[9] CERF, V.G. Beyond the post-PC Internet. **Communications of the ACM**, v.44, n.9, p.34-37. Sep. 2001

[10] ELETROLUX - SCREENFRIDGE. Suécia. Apresenta o protótipo de uma geladeira com capacidade de comunicação via rede. Disponível em: <http://www.electrolux.se/screenfridge>. Acesso em: 09 Jan. 2002.

[11] MACKER, J.P.; CORSON, M.S. Mobile ad hoc networking and the IETF. **Mobile Computing and Communication Review**, v.2, n.1, p.9-14, Jan. 1998.

[12] CORSON, S.; MACKER, J. Mobile Ad Hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations. **IETF Network Working Group Request for Comments**. Jan. 1999. RFC2501.

[13] FREEBERSYSER, James A.; LEINER, Barry. A DoD Perspective on Mobile Ad Hoc Networks. In: PERKINS, Charles E. **Ad hoc networking**. Upper Saddle River, New Jersey: Addison-Wesley, 2000. p.29-52.

[14] ZHOU, L., HAAS, Z.J. Securing Ad Hoc Networks. **IEEE Network**, v.13, i.6, p.24-30, Nov./Dec. 1999.

[15] RUGGIERO, W.V. **Modelo de Segurança para Redes Ad Hoc**. 2002. 107p. Tese (Livre-Docência) – Escola Politécnica, Universidade de São Paulo. São Paulo.

[16] VENTURINI, Y.R. et al. Security model for ad hoc networks. In: INTERNATIONAL CONFERENCE ON WIRELESS NETWORKS – ICWN'02. Las Vegas: CSREA Press, Jun. 2002. **Proceedings**. Las Vegas, Nevada, USA, 2002.

[17] NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY - NIST. **FIPS PUB 140-2: Security Requirements for Cryptography Modules**. Federal Information Processing Standards Publication 140-2. Gaithersburg, Maryland, USA: NIST, May. 2001. 64p.

[18] STALLINGS, W. **Cryptography and network security**: principles and practice. 2.ed. Upper Saddle River, New Jersey: Prentice Hall, 1998. 569p.

[19] STAJANO, F.; ANDERSON, R. The resurrecting duckling: security issues for ad hoc wireless networks. In: AT&T SOFTWARE SYMPOSIUM, 3., Middletown, Oct. 1999. **Proceedings**. New Jersey, USA, 1999.

[20] FEENEY, L.M., AHLGREN, B., WESTERLUND, A. Spontaneous network: an application oriented approach to ad hoc networking. **IEEE Communications Magazine**, v.39, i.6, p.176-181, Jun. 2001.

[21] STAJANO, F. The resurrecting duckling: what next? In: INTERNATIONAL WORKSHOP ON SECURITY PROTOCOLS, 8., Lecture Notes in Computer Science, Springer-Verlag, Santa Barbara, Apr. 2000. **Proceedings**. Santa Barbara, California, USA, 2000.

[22] HAAS, Z.J.; PERLMAN, M. The performance of query control schemes for zone routing protocol. In: ACM SIGCOMM'98, Vancouver, Sep. 1998. **Proceedings**. Vancouver, British Columbia, Canada, 1998.

[23] HAAS, Z.J.; PEARLMAN, M.R.; SAMAR, P. The Interzone Routing Protocol (IERP) for Ad Hoc Networks. **IETF Internet-Draft**. Jun. 2001. Disponível em: <http://www.ietf.org/internet-drafts/draft-ietf-manet-zone-ierp-01.txt>. Acesso em: 24 Apr. 2002.

[24] HAAS, Z.J.; PEARLMAN, M.R.; SAMAR, P. The Intrazone Routing Protocol (IARP) for Ad Hoc Networks. **IETF Internet-Draft**. Jun. 2001. Disponível em: <http://www.ietf.org/internet-drafts/draft-ietf-manet-zone-iarp-01.txt>. Acesso em: 24 Apr. 2002.

[25] PERKINS, C.E.; ROYER, E.M. Ad hoc on demand distance vector routing. In: IEEE WMCSA'99, 2., New Orleans, Feb. 1999. **Proceedings**. New Orleans, Louisiana, USA, 1999.

[26] PERKINS, C.E.; ROYER, E.M.B.; DAS, S.R. Ad hoc On-Demand Distance Vector (AODV) Routing. **IETF Internet Draft**. Jan. 2002.

Disponível em: <http://www.ietf.org/internet-drafts/draft-ietf-manet-aodv-10.txt>. Acesso em: 24 Apr. 2002.

[27] PARK, V.D.; CORSON, M.S. A highly adaptable distributed routing algorithm for mobile wireless networks. In: IEEE INFOCOM'97, Kobe, Apr. 1997. **Proceedings**. Kobe, Japan, 1997.

[28] JONHSON, D.B.; MALTZ, D.A. Dynamic source routing in ad hoc wireless networks. In: IMIELINSKI, T.; KORTH, H.F. **Mobile Computing.** Boston, Massachusetts: Kluwer Academic Publishers, Jan. 1996. p.153.181.

[29] DESMEDT, Y.G.; FRANKEL, Y. Threshold cryptosystems. In: ADVANCES IN CRYPTOLOGY – CRYPTO'89, Lecture Notes in Computer Science, Springer-Verlag, Santa Barbara, Aug. 1989. **Proceedings**. Santa Barbara, California, USA, 1990. p.307-315.

[30] DESMEDT, Y.G. Threshold cryptography. **European Transactions on Telecommunications**, vol.5, n.4, p.449-457, Jul./Aug. 1994.

[31] KÄRPIJOKI, V. Security in ad hoc networks. In: SEMINAR ON NETWORK SECURITY, Helsinki, Fall 2000. **Proceedings**. Helsinki, Finland: Helsinki University of Technology, 2000.

[32] LUNDBERG, J. Routing security on ad hoc networks. In: SEMINAR ON NETWORK SECURITY, Helsinki, Fall 2000. **Proceedings**. Helsinki, Finland: Helsinki University of Technology, 2000.

[33] HUBAUX, J.P. et al. Toward self-organized mobile ad hoc network: the Terminodes project. **IEEE Communications Magazine**, v.39, i.1, p.118-124, Jan. 2001.

[34] BLAZEVIC, L., et al. Self-organization on mobile ad hoc networks: the approach of Terminodes. **IEEE Communications Magazine**, v.39, i.6, p.166-174. Jun. 2001.

[35] FRODIGH, M.; JOHANSSON, P.; LARSSON, P. Wireless ad hoc networks - The art of networking without a network**. Ericsson Review**, n.4 p.248-263, 2000.

[36] GRANDISON, T.; SLOMAN, M. A survey of trust in Internet applications. **IEEE Communications Surveys**, vol.3, n.4, p.2-16, 4th Quarter 2000.

[37] ASOKAN, N. Anonymity in a Mobile Computing Environment. In: IEEE WORKSHOP ON MOBILE COMPUTING SYSTEMS AND APPLICATION, Santa Cruz, 1994. **Proceedings**. Santa Cruz, California, USA, 1994. p.200-204.

[38] VARADHARAJAN, V.; MU, Y. Preserving privacy in mobile communications: a hybrid method. In: IEEE INTERNATIONAL CONFERENCE ON PERSONAL WIRELESS COMMUNICATIONS – ICPWC'97, Bombay, 1997. **Proceedings**. Bombay, India, 1997. p.532-536.

[39] ATENIESE, G. et al. Untraceable mobility: on traveling incognito. **Computer Networks**, v.31, n.8, p.871-884. Apr. 1999.

[40] SOLOMON, J.D. **Mobile IP**: the Internet unplugged. Upper Saddle River, New Jersey: Prentice Hall PTR, 1998. 350p.

[41] BRAY, J.; STURMAN, C.F. **Bluetooth**: connect without cables. Upper Saddle River, New Jersey: Prentice Hall PTR, 2000. 495p.

[42] JAKOBSSON, M.; WETZEL, S. Security weakness in Bluetooth. In: RSA CONFERENCE 2001, San Francisco, Apr. 2001. **Proceedings**. San Francisco, California, USA, 2001. p.176-191.

[43] ASOKAN, N.; GINZBOORG, P. Key agreement in ad hoc networks. **Computer Communications**, v.23, i.17, p.1627-1637, Nov. 2000.

[44] NIETALAHTI, M. Key establishment in ad hoc networks. In: SEMINAR ON NETWORK SECURITY, Helsinki, Fall 2000. **Proceedings**. Helsinki, Finland: Helsinki University of Technology, 2000.

[45] VENKATRAMAN, L.; AGRAWAL, D.P. A novel authentication scheme for ad hoc networks. In: IEEE WIRELESS COMMUNICATION AND NETWORKING CONFERENCE, Chicago, Sep. 2000. **Proceedings**. Chicago, Illinois, USA, 2000. p.1268-1273.

[46] MINGLIANG, J.; LI, J.; TAY, Y.C. **Cluster based routing protocol (CBRP)**. Disponível em: <http://www.math.nus.edu.sg/~mattyc/cbrp.txt>. Acesso em: 20 Apr. 2002.

[47] MANET. List maintained by IETF. Disponível em: <www.ietf.org/mail-archive/working-groups/manet/current/maillist.html>. Acesso em: 24 Apr. 2002.

[48] WEIMERSKIRCH, A.; THONET, G. A distributed light-weight authentication model for ad hoc networks. In: INTERNATIONAL CONFERENCE ON INFORMATION SECURITY AND CRYPTOSYSTEMS – ICISC 2001, 4., Seoul, Dec. 2001. **Proceedings**. Seoul, Korea, 2001.

[49] BALFANZ, D. et al. Talking to strangers: authentication in ad hoc wireless networks. In: NETWORK AND DISTRIBUTED SYSTEM SECURITY SYMPOSIUM – NDSS'02, San Diego, Feb. 2002. **Proceedings**. San Diego, California, USA, 2002.

[50] DIERKS, T.; ALLEN, C. The TLS Protocol Version 1.0. **IETF Network Working Group Request for Comments.** Jan. 1999. RFC2246.

[51] GUPTA, V.; GUPTA, S. Securing the Wireless Internet. **IEEE Communications Magazine**, v.39, i.12, p.68-74. Dec. 2001.

[52] LUO, H., LU, S. **Ubiquitous and robust authentication services for ad hoc wireless networks**. Los Angeles, California, USA: Department of Computer Science, UCLA, Oct. 2000. (Technical Report, TR-200030).

[53] SHAMIR, A. How to share a secret. **Communications of the ACM**, vol.22, n.11, p.612-613, Nov. 1979.

[54] KNUTH, D.E. **The art of computer programming**: volume 2. seminumerical algorithms, 2.ed. Reading, Massachusetts, USA: Addison-Wesley, 1982. Cap.3. p.1-177: Random Numbers.

[55] HELD, G. **Data over wireless networks**: Bluetooth, WAP, & Wireless LANs. Fairfield, Pennsylvania: McGraw Hill Professional Publishing, 2000. 368p.

[56] TORRIERI, D.J. **Principles of Secure Communications Systems**. Norwood, Massachusetts, USA: Artech House, 1985. 453p.

[57] RSA SECURITY INC. **RSA SecurID Authentication**: a better value for a better ROI. RSA Whitepaper. Disponível em: <http:// www.rsasecurity.com/products/securid/>. Acesso em: 15 Aug. 2002.

[58] L'ECUYER, P.; PROULX, R. About polynomial-time "unpredictable" generators. In: WINTER SIMULATION CONFERENCE, Washington, Dec. 1989. **Proceedings**. Washington, D.C., USA, 1989. p.467-476.

[59] CROCKER, S.; SCHILLER, J. Randomness Recommendations for Security. **IETF Networking Work Group Request for Comments**. Dec. 1994. RFC1750.

[60] KELSEY, J.; SCHNEIER, B.; FERGUSON, N. Yarrow-160: Notes on the design and analysis of the Yarrow cryptographic pseudorandom number generator. In: INTERNATIONAL WORKSHOP ON SELECTED AREAS IN CRYPTOGRAPHY, 6., Lecture Notes in Computer Science, Springer-Verlag, Kingston, Aug. 1999. **Proceedings**. Kingston, Ontario, Canada, 1999. p.13-33.

[61] KELSEY, J. et al. Cryptanalytic attacks on pseudorandom number generators. In: INTERNATIONAL WORKSHOP ON FAST SOFTWARE ENCRYPTION, 5., Lecture Notes in Computer Science, Springer-Verlag, Paris, Mar. 1998. **Proceedings**. Paris, France, 1998. p. 168-188.

[62] BLUM, L.; BLUM, M. SCHUB, M. Comparison of two pseudo-random number generators. In: ADVANCES IN CRYPTOLOGY – CRYPTO'82, Lecture Notes in Computer Science, Springer-Verlag, Santa Barbara, Aug. 1982. **Proceedings**. Santa Barbara, California, USA, 1982. p.61-78.

[63] BLUM, L.; BLUM, M. SCHUB, M. A simple unpredictable pseudo-random number generator. **SIAM Journal of Computing**, v.15, n.2, p.364-383, May 1986.

[64] AMERICAN BANKERS ASSOCIATION - ABA. **Digital signatures using reversible public key cryptography for the financial services industry (rDSA)**. ANSI X9.31. 1998. App.A.2.4.

[65] NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY - NIST. **FIPS 186-2: Digital Signature Standard (DSS)**. Federal Information Processing Standards Publication 186-2. Gaithersburg, Maryland, USA: NIST, Jan. 2000. App.3. p.16-19: Random number generation for the DSA.

[66] RSA LABORATORIES. **RSAREF: a cryptographic toolkit**. version 2.0. RSA Data Security, Inc, Mar. 1994. Disponível em: <ftp://ftp.funet.fi/pub/crypt/cryptography/asymmetric/rsa/rsaref2.tar.gz>. Acesso em: 15 Jun. 2002.

[67] MILLS, D.L. Network Time Protocol (version 3): Specification, implementation and analysis. **IETF Network Working Group Request for Comments**. Mar. 1992. RFC1305.

[68] DIFFIE, W.; HELLMAN, M.E. New directions in cryptography. **IEEE Transactions on Information Theory**, v.22, n.6, p.644-654, Nov. 1976.

[69] ARNOLD, K. et al. **The Jini™ Specification**. Reading, Massachusetts, USA: Addison Wesley, 1999. 385p.

[70] MICROSOFT CORPORATION. **Universal Plug and Play Device Architecture**. version 1.0. Jun. 2000. Disponível em: <http://www.upnp.org/resources/documents.asp>. Acesso em: 13 Mar. 2002.

[71] MATAYOSHI, C.M. **Modelo de Segurança da Linguagem Java: Problemas e Soluções**. 2001. 111p. Dissertação (Mestrado) – Escola Politécnica, Universidade de São Paulo. São Paulo.

[72] WINDOWS PACKET CAPTURE LIBRARY. Torino, Italia. 2002. Descreve e disponibiliza o pacote WinPcap. Politecnico di Torino. Disponível em: <http://winpcap.polito.it>. Acesso em: 16 Aug. 2002.

[73] POSKANZER, J. Berkeley, California, USA. 1996. Disponibiliza o pacote Acme.Crypto. Disponível em: <http://www.acme.com>. Acesso em: 7 Jul. 2002.

[74] MENEZES, A.J.; OORSCHOT, P.C.V.; VANSTONE, S.A. **Handbook of Applied Cryptography**. Boca Raton, Florida, USA: CRC Press, 1996. 816p.

[75] JAIN, R. **The Art of Computer Systems Performance Analysis**: Techniques for experimental design, measurements, simulation, and modeling. New York, New York, USA: John Wiley & Sons, 1991. 720p.

[76] NETO, P.L.O.C. **Estatística**. São Paulo, São Paulo, Brasil: Edgard Blücher, 1977. 260p.

Bibliografia Recomendada

BISDIKIAN, C.; BHAGWAT, P.; GOLMIE, N. Wireless personal area networks. **IEEE Network**, v.15, i.5, p.10-11, Sep.-Oct. 2001.

VANHALA, A. Security in ad hoc networks. In: RESEARCH SEMINAR ON SECURITY IN DISTRIBUTED SYSTEMS. Helsinki, Nov. 2000. **Proceedings**. Helsinki, Finland: University of Helsinki, 2000.

BORISOV, N.; GOLDBERG, I.; WAGNER, D. Intercepting mobile communications: the insecurity of 802.11. In: ANNUAL INTERNATIONAL CONFERENCE ON MOBILE COMPUTING AND NETWORKING – MOBICOM'01, 7., Rome, Jul. 2001. **Proceedings**. Rome, Italy, 2001.

MILLER, S.K. Facing the challenge of wireless security. **IEEE Computer**, v.34, i.7, p.16-18. Jul. 2001.

KORBA, L. Security system for wireless local area networks. In: INTERNATIONAL SYMPOSIUM ON PERSONAL, INDOOR AND MOBILE RADIO COMMUNICATIONS - PIMRC'98, 9., Boston, Sep. 1998. **Proceedings**. Boston, Massachusetts, USA, 1998.

ERONEN, P.; GEHRMANN, C.; NIKANDER, P. Securing ad hoc Jini services. In: NORDIC WORKSHOP ON SECURE IT SYSTEMS – NORDSEC'00, 5., Reykjavik, Oct. 2000. **Proceedings**. Reykjavik, Iceland: Reykjavik University, 2000. p.169-177.

GEHRMANN, C.; NIKANDER, P. Securing ad hoc services, a Jini view. In: IEEE FIRST ANNUAL WORKSHOP ON MOBILE AND AD HOC NETWORKING AND COMPUTING - MOBIHOC'00, Boston, Aug. 2000. **Proceedings**. Boston, Massachusetts, USA, 2000. p.135-136.

SEO, D.H.; SWEENEY, D. Simple authenticated key agreement algorithm. **IEE Electronic Letters**, v.35, n.13, p.1073-1074. 24th Jun. 1999.

TSENG, Y.M. Weakness in simple authenticated key agreement protocol. **IEE Electronic Letters**, v.36, n.1, p.48-49. 6th Jan. 2000.

AZIZ, A.; DIFFIE, W. Privacy and authentication for wireless local area networks. **IEEE Personal Communications**, v.1, i.1, p.25-31, 1st Qtr. 1994.

XU, S.; SAADAWI, T. Does the IEEE 802.11 MAC protocol works well in multihop wireless ad hoc networks? **IEEE Communications Magazine**, v.39, i.6, p.130-137, Jun. 2001.

BRUNO, R.; CONTI, M.; GREGORI. E. WLAN technologies for mobile ad hoc networks. In: HAWAII INTERNATIONAL CONFERENCE ON SYSTEMS SCIENCES, 34., Maui, Jan. 2001. **Proceedings**. Maui, Hawaii, USA, 2001.

LANSFORD, J.; STEPHENS, A.; NEVO, R. Wi-Fi (802.11b) and Bluetooh: enabling coexistence. **IEEE Network**, v.15, i.5, p.20-27, Sep./Oct. 2001.

MULLER, N.J. **Bluetooth Demystified**. New York, New York: McGraw-Hill TELECOM, 2000. 396p.

MILLER, B.A.; BISDIKIAN, C. **Bluetooth revealed**: the insider's guide to an open specification for global wireless communications. Upper Saddle River, New Jersey: Prentice Hall PTR, 2000. 303p.

L'ECUYER, P. Uniform random number generator: a review. In: ACM WINTER SIMULATION CONFERENCE, Atlanta, Dec. 1997. **Proceedings**. Atlanta, Georgia, USA, 1997. p.127-134.

SCHNEIER, B. **Applied Cryptography**. 2.ed. New York, New York, USA: John Wiley & Sons, 1995. 784p.

IEEE COMPUTER SOCIETY. LAN/MAN Standards Committee. **ANSI/IEEE 802.11a Std**. Supplement to ANSI/IEEE Std. 802.11. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: High-speed Physical Layer in the 5 GHz Band. New York, New York, USA: IEEE, Sep. 1999. 83p.