
**Pontifícia Universidade Católica de Campinas**

**CEATEC – Centro de Ciências Exatas, Ambientais e de Tecnologia**

**Análise de Sistemas**

# Proposta de Sistema Anti-Invasão

**Disciplina de Redes e Comunicação de Dados II**

**Professor André BZ**

**Equipe**:

| | | |
|---|---|---|
| Daniela Silveira | 02002897 | dc_silveira@yahoo.com.br |
| Flávio Cortez | 03315082 | flaviocortez@superig.com.br |
| Fábio Vasconcelos | 01095173 | fabiovas@br.ibm.com |
| Felipe Barbi | 02022812 | febarbi@terra.com.br |
| Guilherme Seleguim | 02020519 | cestarolli@infinito.it |
| Rody Vasconcelos | 01076918 | rodyps@br.ibm.com |

# Resumo

Com as inovações tecnológicas as ameaças digitais evoluem e estão cada vez mais inteligentes e poderosas. A rapidez com que se propagam e o montante do prejuízo que causam são números para se considerar. Este documento foi elaborado com o intuito de fornecer uma proposta de sistema anti-invasão baseado nas últimas ameaças digitais que foram divulgadas. O estudo para um sistema anti-invasão foi voltado para as ameaças do tipo ***backdoor/trojan horse*** (porta dos fundos/cavalo-de-tróia), porém, algumas proteções aqui discutidas podem abranger também **vírus**, que incorporem certas funcionalidades. A proposta apresentada não almeja ser uma única solução, ou uma solução definitiva para o tema "segurança da informação digital", nem tão pouco ser comparada à um sistema **firewall** ou sistema antivírus. A proposta visa, portanto, minimizar as ações e propagações das ameaças, evitando maiores prejuízos. O **Ice Eyes Security Agent** pode ser baixado em http://ice-eyes.cjb.net.

# Palavras-chave

Segurança da Informação, *Hackers*, *Crackers*, Cavalo-de-Tróia, Vírus, Redes de Computadores, Invasão Computacional, Anti-Invasão.

# *Anti-Invasion System Proposal*

***Systems Analysis***

***Computer Networks and Data Comunication II***

***Professor André B. Z.***

# *Summary*

*With technological innovations the digital threats are progressing and are more smarts and powerfull. The quickness that it propagates and the damages that it generates are numbers to consider . This document was made to be an anti-invasion system proposal based on the last recently known digital threats. The study for an anti-invasion system was made based on threats like backdoor / trojan horse, but, some protections discussed here can be for some kinds of vírus that incorporate certain functionallyties. The proposal shown here doesn't want to be the only solution, or a definitive solution for the theme "digital information security", or to be compared as an firewall system or as an antivirus system. The proposal is to reduce the actions and propagations of the threats, preventing big damages to happen.**The Ice Eyes Security Agent** can be downloaded at http://ice-eyes.cjb.net.*

## *Keywords*

*Information Security, Hackers, Crackers, Trojan Horses, Vírus, Computer Networks, Systems Invasion, Anti-Invasion.*


# 1. Introdução

Dentre as últimas cinquenta ***threats***[1] que foram divulgadas por empresas de segurança da informação no decorrer deste projeto, quarenta eram do tipo ***backdoor/trojan horse***[2]. Não é de se admirar que a obtenção de acesso remoto ao maior número de ***hosts***[3] possíveis é um objetivo bastante almejado, porém não tão difícil de ser alcançado atualmente.

Desde a criação do primeiro ***worm***[4] por ***Robert Tappan Morris***[5] em 03 de Novembro de 1988 a preocupação com a segurança digital vem crescendo a cada ano, e as ameaças com maiores recursos de inteligência e disseminação.

O ***worm*** chegou até a ser comparado com a **AIDS** pela imprensa da época, sugerindo que colocaria um fim na chamada "**Era da Informação**". Mais de uma década depois ainda não estamos seguros e segundo dados fornecidos pela **Nona Pesquisa Nacional de Segurança da Informação**[6] desenvolvida pela **Módulo Security**[7], há uma tendência de aumento quanto às ameaças, os riscos e os ataques no ano de 2004. 32% dos entrevistados nesta pesquisa apontam os ***Hackers***[8] / ***Crackers***[9] como os principais responsáveis por ataques e **invasões**[10] aos sistemas corporativos e 60% indicam a Internet como principal ponto de invasão. O vazamento de informações foi apontado como a ameaça mais crítica para o negócio das corporações. Cada empresa foi vítima de aproximadamente 38 ataques por semana no primeiro semestre de 2003 e os prejuízos chegam a mais de um milhão de reais, mas nota-se também que na maioria dos casos as empresas não conseguem quantificar o prejuízo.

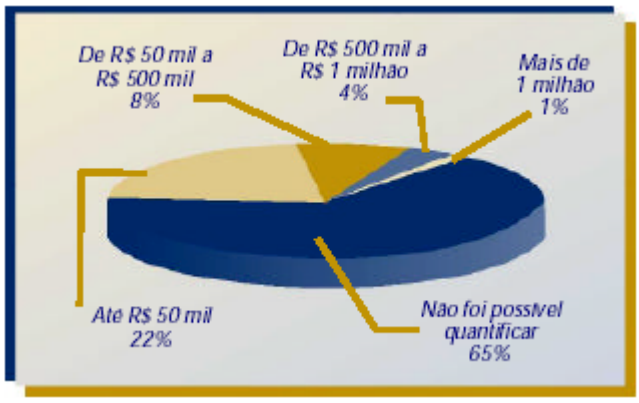



**Fig.01 Prejuízos com ataques (Fonte: 9º PNSI).**

Os principais responsáveis por ataques e invasões são os ***Hackers*** mas grande

1. Ameaças digitais, i. e., software desenvolvido com a possibilidade de causar danos à ambientes e sistemas computacionais.

2. Porta dos fundos / cavalo-de-tróia, i. e., software para acesso remoto. Seu nome é originado da lendária história da guerra entre gregos e troianos, representada na obra "Ilíada" de Homero.

3. Aparelho / máquina conectada à uma rede de conputadores.

4. Software que faz cópias de si mesmo, por exemplo, de um disquete para um HD, ou se copiando em e-mails ou outro mecanismo de transporte. Um ***worm*** pode causar dano e comprometer a segurança de um computador, podendo chegar escondido dentro de algum software.

5. Na condição de filho do principal encarregado do ***National Computer Security Center***, rapaz inteligente e com futuro promissor, Robert ficou conhecido nos meios jornalísticos por contaminar a Internet, propositalmente, com um vírus de nome "***Worm***" (minhoca). Em pouco tempo, inúmeros computadores foram infectados e entraram em pane. Em seu julgamento, Robert afirmou que a contaminação não tinha sido intencional, mas pagou seu erro com uma pena relativamente rigorosa, sendo condenado a cinco anos de prisão com direito a liberdade condicional. Hoje, os ***worms*** são largamente disseminadas através da Internet, propagando-se e contaminando computadores em massa, principalmente via e-mail.

6. Pesquisa realizada pela Módulo Security para avaliar a segurança da informação em âmbito nacional.

7. Empresa voltada à segurança da informação.

8. Especialista em computadores

que usa seus conhecimentos e habilidades para invadir sistemas, obtendo acesso a informação. Normalmente não danifica hardware / software em suas empreitadas, deixando esta atividade para os ***Crackers***.

9. Oposto do ***Hacker***, suas habilidades e conhecimentos são exclusivamente voltadas para danificar equipamentos computacionais físicos e lógicos.

10. Obter acesso não autorizado a sistemas computacionais privados, tanto remotamente como localmente, comprometendo assim a integridade dos dados, mesmo que sem fazer alterações. Muitas empresas não divulgam o fato de terem sido invadidos com medo de prejudicar sua imagem perante os seus clientes e a sociedade.

11. Estudo realizado pelo *CSI-Computer Security Institute / FBI-Federal Bureau of Investigation* voltado à segurança da informação.

parte das empresas não consegue identificar a causa dos ataques e invasões.

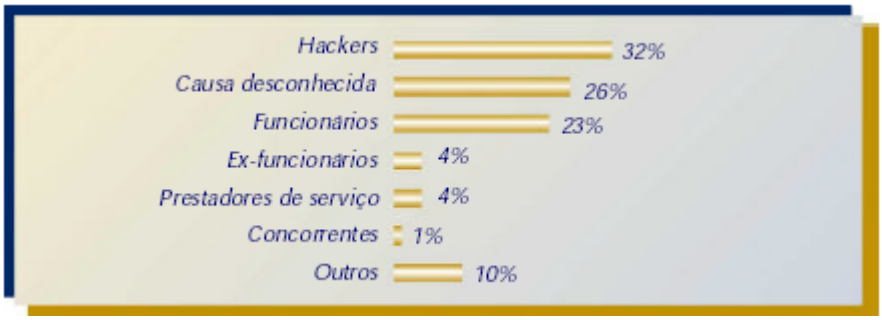



**Fig. 02 Principais responsáveis por ataques e invasões (Fonte: 9º PNSI)**

Embora os ataques identificados como sendo de Concorrentes não ultrapasse a faixa de 1%, esta possibilidade não pode ser deixada de fora ou esquecida. O vazamento ou roubo de informação, mesmo que na forma não digital, é um fator preocupante. Um acesso indevido à uma base de dados pode significar anos de trabalho e informação perdidos, podendo nunca mais serem recuperados.

A maior parte das empresas indica a Internet como o principal ponto de invasão em seus sistemas. No estudo ***Computer Crime and Security 2003***[11] cerca de 78% dos entrevistados apontaram a Internet como o principal ponto de invasão em transações eletrônicas.

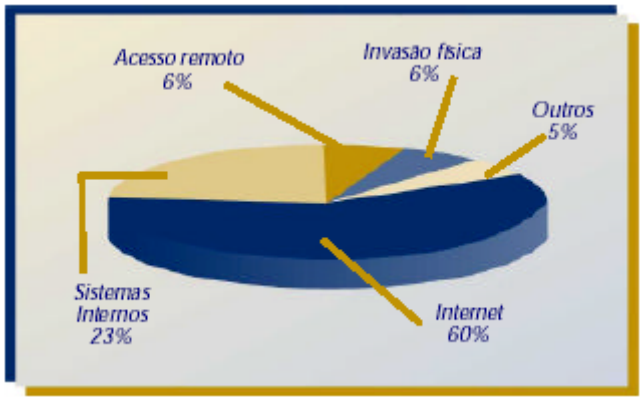



**Fig. 03 Principais pontos de invasão (Fonte: 9º PNSI).**

# 2. Digital Threats

A **Teoria Geral de Sistemas**[12] ensina que, todo sistema, mesmo que construído beneficamente(sem nenhum código malicioso ou falta de ética do desenvolvedor), tendem à **entropia**[13], mesmo que em estado de **homeostasia**[14], independentemente de sua complexidade. Sofrem, portanto, influências do meio ao qual se encontram, influências estas que agem diretamente no funcionamento dos sistemas. As ameaças digitais fazem parte do mundo virtual, assim como assaltantes e terroristas compõem o mundo real. Dependendo do caminho que percorra, algumas ameaças poderão surgir pela frente ou mesmo que não vá para lugar algum, elas poderão vir até você. As ameaças evoluem com as inovações tecnológicas e ficam cada vez mais poderosas e dinâmicas. "*O pensamento imita a ação e a ação imita o pensamento*", portanto não podemos nos esquecer que este tipo de software não é criado sozinho e há sempre uma, ou várias, cabeças pensantes por trás dos monitores, em algum lugar do mundo, construindo estes softwares. Em **1949**, no Instituto de Estudos Avançados em **Princeton**, New Jersey, em um documento intitulado ***"Theory and Organization of Complicated Automata"***, **John von Neumann**[15] propôs que era teoricamente possível que um programa de computador se replicasse. O documento inclui um modelo do que hoje é conhecido como vírus de computador.

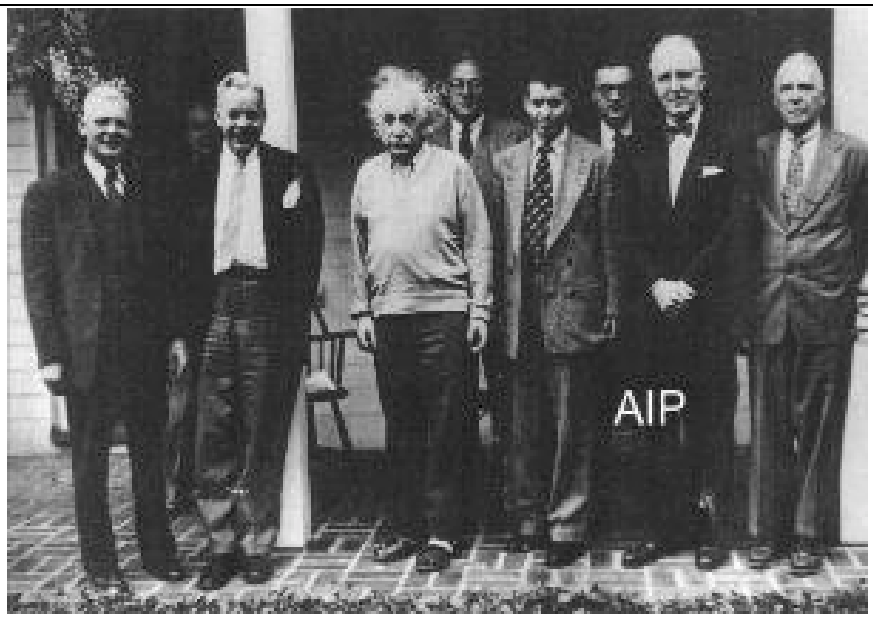
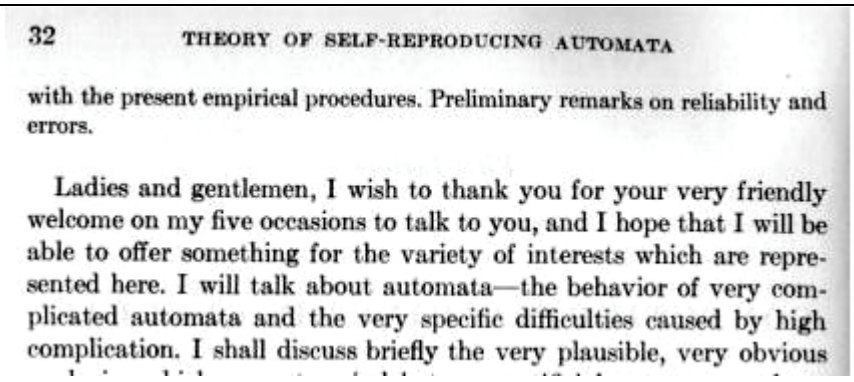



**Fig.04 John Von Neumann, Albert Einstein e outros senhores em Princeton (Fonte: Emilio Segrè Visual Archieves - http://www.aip.org/history/esva/ html/einstein_e34.html).**



32 THEORY OF SELF-REPRODUCING AUTOMATA

with the present empirical procedures. Preliminary remarks on reliability and errors.

Ladies and gentlemen, I wish to thank you for your very friendly welcome on my five occasions to talk to you, and I hope that I will be able to offer something for the variety of interests which are represented here. I will talk about automata—the behavior of very complicated automata and the very specific difficulties caused by high complication. I shall discuss briefly the very plausible, very obvious

**Fig.04 Introdução à teoria, "*Theory and Organization of Complicated Automata*", *John Von Neumann* (Fonte: http://www.walenz.org/vonNeumann/).**

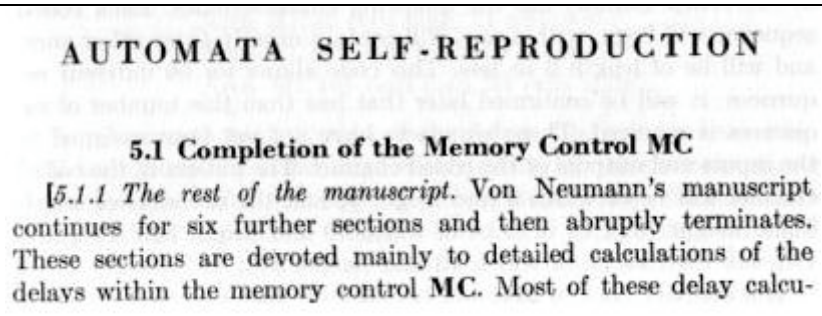
12. Disciplina do Curso de Análise de Sistemas. A Teoria Geral de Sistemas demonstra que as ciências podem ser vistas de uma mesma forma (isomorfismo), permitindo maior aproximação entre as fronteiras das ciências e o preenchimento dos espaços vazios entre elas. Com a Teoria Geral de Sistemas, os diversos ramos do conhecimento – até então estranhos uns aos outros pela intensa especialização e isolamento conseqüente – passam a tratar seus objetos de estudo como sistemas. Ex: Sistema Físico, Sistema Químico, Sistema Biológico, Sistema Psíquico, Sistema Social, Sistema Empresa, Ecossistema, etc.

13. Tendência ao mal funcionamento ou não funcionamento, tendência à parar de funcionar. Entropia vem do grego "*entrope*" que significa uma transformação e foi originalmente conceituada na segunda lei da termodinâmica e refere-se à perda de energia em sistemas isolados, levando-os à degradação, à desintegração e ao seu desaparecimento. Na Teoria Geral de Sistemas, a entropia é a tendência que os sistemas tem para o desgaste, para a desintegração, para o afrouxamento dos padrões e para um aumento da aleatoriedade. Se a entropia significa a tendência à perda, à desintegração e à desorganização, o reverso da entropia é a entropia negativa ou negentropia. A negentropia é o suprimentoda informação adicional capaz, não apenas de repor as perdas das mesmas, mas também, de proporcionar a integração e a organização no sistema.

14. Estado de normalidade de sistemas, funcional. O conceito de homeostase ou homeostasia nasceu na fisiologia animal com Claude Bernard (1813-

1878) que afirmou que "*Todos os mecanismos vitais têm por objetivo conservar constantes as condições de vida no ambiente interno*" e que "*A estabilidade do meio interno é a condição primordial da vida livre*". Cada função do corpo é cercada por seu meio, o qual é importante não só para seu funcionamento como para sua integridade. Em 1929, Walter B. Cannon (1871-1945) ampliou o conceito de meio interior com a noção de homeostasia (grego *homeos*=semelhante e *statis*=situação. Cada parte do organismo funciona normalmente em um estado de equilíbrio. A homeostase é um equilíbrio dinâmico obtido através da auto-regulação ou seja através do auto-controle. É a capacidade que tem o sistema de manter certas variáveis dentro de limites, mesmo quando os estímulos do meio externo forçam estas variáveis a assumirem valores que ultrapassam os limites da normalidade. Os seres humanos vivem através de um processo contínuo de desintegração (entropia) e de reconstituição dentro do ambiente (homeostase). Se o equilíbrio homeostático não resistir àquele fluxo de desintegração e corrupção, o ser humano começará a se desintegrar, mais do que pode se reconstruir, e assim, morrerá.

15. John von Neumann foi um dos matemáticos mais notáveis de nossos tempos. Como tantos outros matemáticos ele prestou contribuições importantes tanto à ciência quanto à matemática. Von Neumann se sentia particularmente fascinado pelos jogos de estratégia e de acaso. Assim, não é de se surpreender, que fosse ele uma das pessoas que abrisse o novo campo da matemática chamado *teoria dos jogos*. Empregando as probabilidades envolvidas em um jogo de acaso e trabalhando com estratégias que produzem "vencedores" em jogos de



AUTOMATA SELF-REPRODUCTION

**5.1 Completion of the Memory Control MC**

[*5.1.1 The rest of the manuscript.* Von Neumann's manuscript continues for six further sections and then abruptly terminates. These sections are devoted mainly to detailed calculations of the delays within the memory control **MC**. Most of these delay calcu-

**Fig.05 Começo da explicação de auto-reprodução, "*Theory and Organization of Complicated Automata*", *John Von Neumann* (Fonte: http://www.walenz.org/vonNeumann/).**

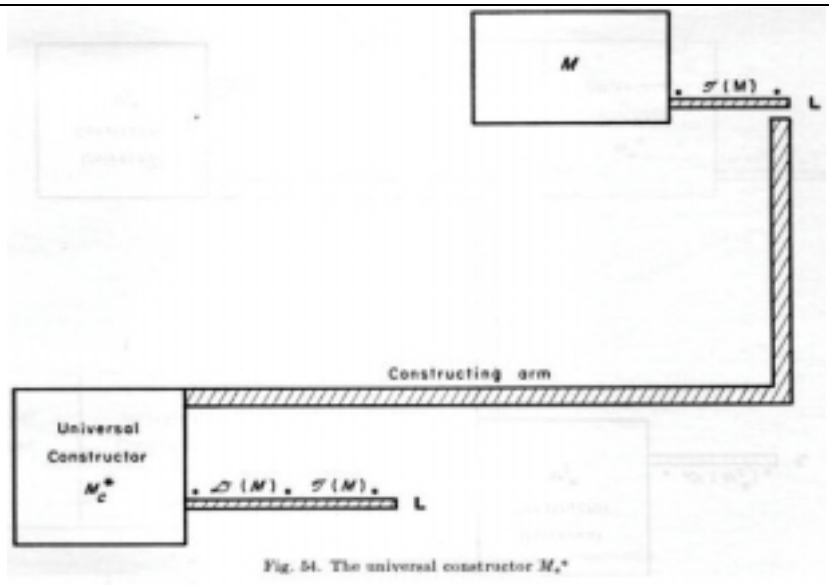



Fig. 54. The universal constructor $M_c^*$

**Fig.06 *The universal constructor, "Theory and Organization of Complicated Automata", John Von Neumann* (Fonte: http://www.walenz.org/vonNeumann/).**

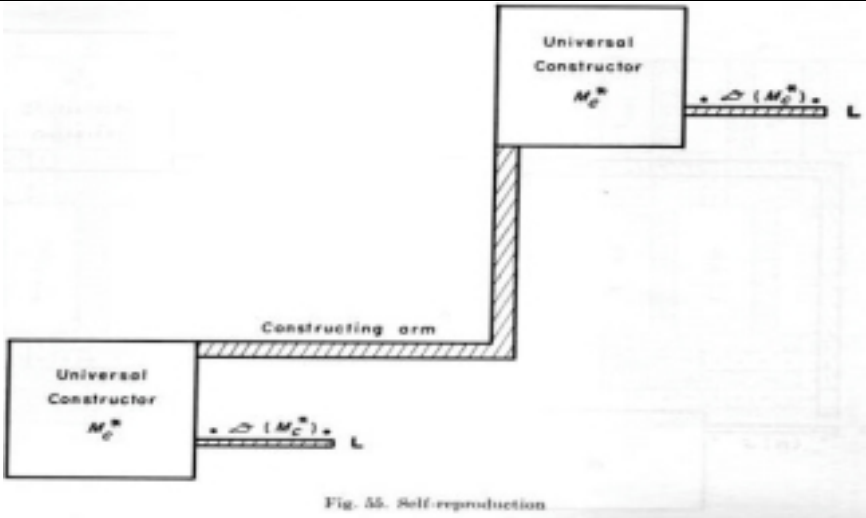



Fig. 55. Self-reproduction

**Fig.07 *Self-reproduction, "Theory and Organization of Complicated Automata", John Von Neumann* (Fonte: http://www.walenz.org/vonNeumann/).**

tomar decisões, a teoria dos jogos de Von Neumann pode solucionar problemas de economia, de ciência e de estratégia militar. Von Neumann nasceu em Budapeste, na Hungria. Aos seis anos era capaz de resolver mentalmente problemas de divisão como 78.463.215 ÷ 49.673.235. Por volta dos oito anos, obteve seu diploma de cálculo na faculdade e como brincadeira podia memorizar, apenas olhando, os nomes, os endereços e números de telefone de uma coluna em uma lista telefônica. Com apenas 23 anos escreveu um livro chamado *Os fundamentos matemáticos da mecânica quântica*, utilizado no desenvolvimento da energia atômica.

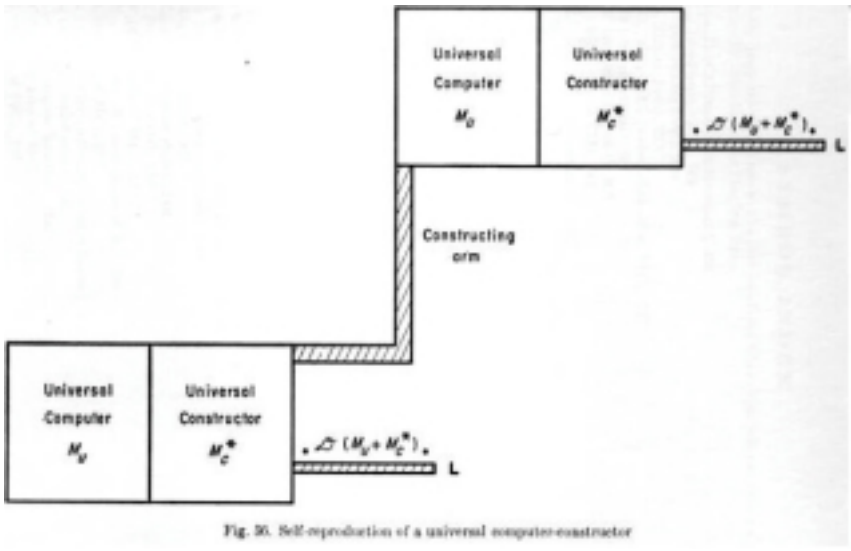



**Fig.08** ***Self-reproduction of a universal computer-constructor, "Theory and Organization of Complicated Automata", John Von Neumann*** **(Fonte: http://www.walenz.org/vonNeumann/).**

16. Empresa voltada para pesquisas e estudos para tecnologia das comunicações

Na década de 60 surgiriam os primeiros antecessores dos vírus atuais. Nesta década, **H. Douglas McIlroy**, **Victor Vysottsky**, e **Robert Morris**, programadores da ***Bell Laboratories***[16], desenvolveram um jogo chamado "***Core Wars***", que era capaz de se reproduzir toda vez que era executado, sobrecarregando assim a memória do equipamento de outro jogador. Nessa mesma época, os criadores deste peculiar jogo também foram responsáveis pela criação do primeiro antivírus, um aplicativo chamado "***Reeper***", que era capaz de destruir cópias criadas pelo "***Core Wars***".

17. Portal sobre segurança da informação

18. Empresa mundialmente conhecida pelos seus produtos de software, sendo um deles o Sistema Operacional Windows.

19. *Local Security Authority Subsystem Service*, controla várias tarefas de segurança do Windows, incluindo o acesso ao sistema.

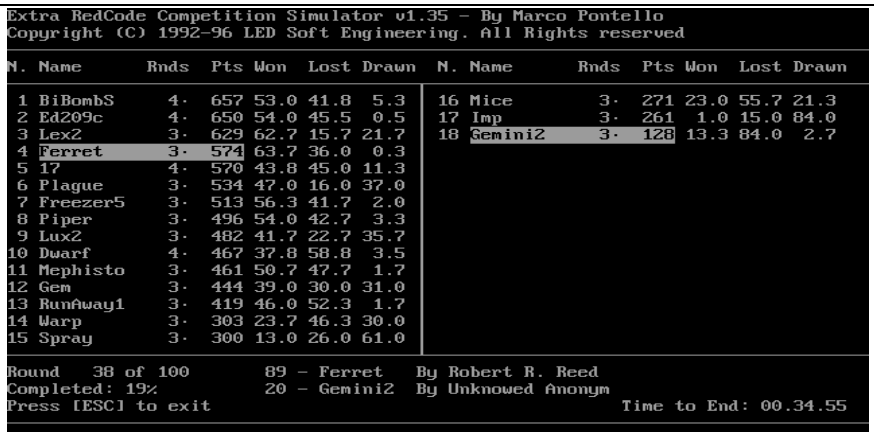



**Fig.09** ***Core Wars Screen Shot*** **(Fonte: http://mark0.ngi.it/corewars/xrk/XRC-Running.png).**

20. Vírus de computador descoberto em Agosto de 2003 que se utiliza de uma falha no protocolo *Remote Procedure Call*(*RPC*).

Depois de mais de 50 anos da teoria original de **Neumann** as ameaças evoluíram consideravelmente. A empresa antivírus **Sophos** divulgou em seu relatório mensal que o **Netsky** foi o vírus mais ativo de abril. Sete versões diferentes do vírus foram responsáveis por 69% das ocorrências de códigos maléficos relatadas à empresa durante o mês de Abril de 2004. No mês de Março, outras variantes do **Netsky** também ocuparam os primeiros lugares no ranking das 10 pragas mais ativas.

21. Compilador desenvolvido pela Microsoft Corporation.

22. Portas identificam serviços que rodam em servidores. Um servidor pode conter diversos

A versão mais relatada foi o **Netsky.P**, com 23% dos ataques, seguida pelo **Netsky.B**, com 20%. A variante D aparece em terceiro, com 17% dos incidentes. Aparecem ainda na lista dos dez piores vírus o **Sober.F**, o **Bagle.Zip** e o **Gibe.F**. A **Sophos** identificou 740 novos vírus em Abril.

"*Durante todo o mês de abril, diversas variantes do vírus **Netsky** continuaram a causar sérios problemas aos usuários de computadores desprotegidos. Uma vez que o autor do **Netsky** original afirma ter aberto o código malicioso, é possível que outros gatunos tenham se aproveitado para enviar novas variantes do **Netsky***", afirmou a consultora de segurança da **Sophos**, Carole Theriault.

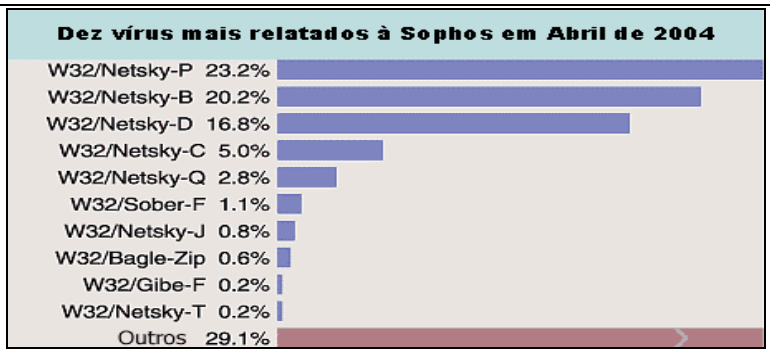



**Fig.10 Top 10 Vírus de Abril de 2004. (Fonte: InfoGuerra).**

Para exemplificar a inovação tecnológica, inteligência e poder de propagação das novas ameaças virtuais, utilizaremos para análise o ***worm*** "**Sasser**", originalmente batizado de **W32/Sasser.A**. Conforme notícia divulgada através do portal **InfoGuerra**[17] no dia 01/05/2004, foi anunciada a descoberta do primeiro ***worm*** projetado exclusivamente para se aproveitar de falhas em componentes do Windows, divulgadas (e corrigidas) no mais recente boletim de segurança da **Microsoft**[18], o **MS04-011**, um dos mais críticos dos últimos meses. O código do ***worm*** distribui-se automaticamente em redes de computadores aproveitando-se de uma falha no serviço **LSASS**[19].

O ***worm*** **Sasser** tem várias semelhanças com o **Blaster**[20], surgido em agosto do ano passado. O aparecimento de ambos era esperado após a divulgação de falhas no Windows. Assim como o **Blaster**, o **Sasser** afeta Windows 2000 e XP e pode travar ou reiniciar os sistemas com um aviso na tela. Uma cópia destes avisos disponibilizada no site da empresa antivírus ***F-Secure*** é mostrada abaixo:

serviços instalados, ou seja, o mesmo computador pode ser um servidor de correio eletrônico, servidor FTP e servidor Web (WWW). O servidor é identificado por um endereço IP, mas os serviços também precisam ser identificados individualmente. Para cada serviço é associada uma porta: um número de identificação entre 0 e 65535. Os serviços padronizados recebem números (portas) padronizadas. São as chamadas portas conhecidas ou *well-known ports*. Esses números são definidos pelo IANA(*Internet Assigned Numbers Authority*) e estão na faixa de 1 a 1024, Essa padronização permite aos clientes conectarem-se a serviços, uma vez que eles já sabem em que porta encontrá-lo. Por exemplo, um servidor Web normalmente aguarda requisições dos *browsers* na porta 80. O *browser* irá se conectar ao servidor Web através dessa porta do servidor. O computador cliente também utilizará uma porta, pois o servidor Web terá que enviar as páginas solicitadas por essa porta. O sistema operacional do cliente determinará, dinamicamente, um número maior que 1024 para a porta do cliente. Essas portas também são conhecidas como portas alocadas dinamicamente.

23. Sigla que significa "*Transmission Control Protocol*", ou Protocolo de Controle de Transmissão. O protocolo TCP é um serviço de entrega de pacotes que garante a entrega e a integridade e funciona baseado na conexão lógica entre dois computadores. Nesse tipo de conexão, ambas as partes entram em acordo sobre a forma de trocarem informações. Quando uma informação é transmitida, mecanismos de verificação de erros e controle de fluxo garantem que a informação seja recebida sem erros. O TCP é utilizado na transmissão

de dados críticos, onde nenhum erro é aceitável. Muitos programas e protocolos de rede tais como HTTP e FTP utilizam o TCP. Antes de transmitir os dados, o protocolo TCP estabelece uma conexão entre os computadores, num processo chamado "*three-way handshake*" (apresentação de três vias). Ao final da transmissão, a conexão é fechada através do mesmo processo. Para transmitir os dados, o pacote TCP é dividido em segmentos que chegaram fora de ordem ou solicitando o reenvio de segmentos que não foram recebidos. Cada segmento é verificado, através de um "*checksum*", para garantir que não tenha sido corrompido na transmissão por erros ou interferências no meio físico.

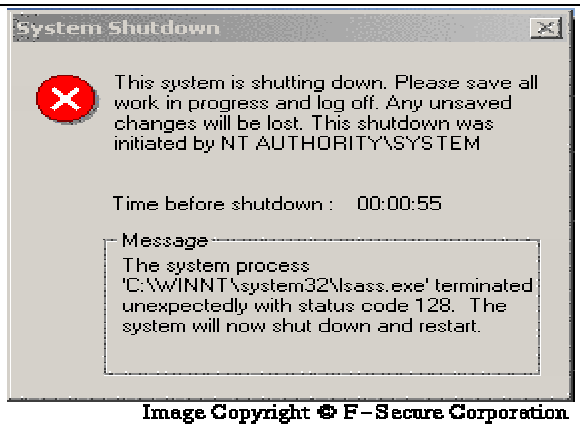



**Fig.11 Mensagem produzida sob Windows 2000.**


**Fig.12 Mensagem produzida sob Windows XP.**

**Sasser** é um código escrito em **Visual C++**[21] que se envia por redes usando três **portas**[22] **TCP**[23] no processo: 445, 9996 e 5554. Pela **porta** 445, o ***worm*** varre a rede, a partir de uma máquina infectada, em busca de outras máquinas cuja vulnerabilidade do serviço **LSASS** não tenha sido corrigida. É nesta fase que os sistemas vulneráveis, ao serem atingidos, podem travar e desligar.

Se o ataque for bem-sucedido, o ***worm*** inicia uma shell (tela de acesso) na **porta** 9996, através da qual o sistema é instruído a fazer uma conexão **FTP**[24] para baixar e executar o código final do ***worm***. Este servidor **FTP** fica disponível na porta 5554 em todas as máquinas infectadas e é usado pra distribuir o código maléfico para outras máquinas que estão sendo contaminadas. Toda a atividade do servidor **FTP** é guardada no arquivo C:\win.log, que serve como indício de que um sistema foi infectado pelo ***worm***.

Ao se instalar no sistema, o **Sasser** cria uma cópia de si mesmo no diretório do **Windows**, com o nome "avserve.exe". Este mesmo nome é adicionado à seguinte chave do **Registro**[25]:

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run

24. Sigla que significa "*File Transfer Protocol*", ou Protocolo de Transferência de Arquivos. O protocolo FTP permite a transferência de arquivos entre um computador local e um servidor remoto, sendo bastante utilizado para *upload* e *download* de arquivos de sites na Internet. A implementação da aplicação FTP é baseada na arquitetura Cliente-Servidor. É necessária a existência de um servidor FTP no computador remoto e de um cliente no computador local. O servidor FTP aguarda conexões de clientes na porta 21 e o cliente utiliza qualquer número de porta local. O protocolo FTP permite a navegação em uma parte da estrutura de diretórios do servidor remoto para a localização do arquivo desejado. No início da conexão com o servidor remoto é solicitada a digitação de um nome de usuário e uma senha, que serão utilizados para validá-lo e determinar seus direitos de acesso. No caso da Internet, alguns sites possuem arquivos ou diretórios confidenciais, que somente podem ser acessados por usuários autorizados. Entretanto

O ***worm*** também cria um ***mutex***[26] com nome de "Jobaka3l".

Para se proteger, usuários de **Windows** devem atualizar seus sistemas no site **Windows Update**.

Apesar do **Sasser** ser considerado o primeiro ***worm*** criado especificamente para explorar as falhas do **Windows** divulgadas recentemente, outros tipos de ***malware*** já vinham se aproveitando das mesmas falhas.

No dia 03/05/2004 foi identificado que o **Sasser** possuía duas novas versões. A versão B é considerada de alto risco e já se encontra em larga distribuição, de acordo com a **Trend Micro**[27]. A **Symantec**[28], que inicialmente havia classificado o **Sasser.B** como de risco 3, **elevou** esse risco para 4, numa escala que vai até 5.

Assim como a primeira versão, o **Sasser.B** vasculha redes em busca de sistemas **Windows** cujas falhas não tenham sido corrigidas. No processo de disseminação do ***worm*** são usadas as **portas TCP** 445, 9996 e 5554.

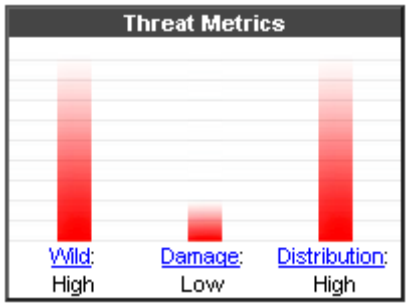



**Fig.13 Threat Metrics para Sasser.B (Fonte: Symantec Security Response).**

A diferença em relação à versão original está nos nomes dos arquivos descarregados pelo **Sasser.B** -- win2.log e avserve2.exe, em vez de win.log e avserve.exe -- e o fato de que o ***worm*** abre 128 processos em vez de 128 segmentos ("*threads*" ou "subprocessos"). O resultado é que, se um processo "travar", não afetará os outros. No caso de segmentos, quando um trava todos os outros também podem ser afetados. A abordagem presente na versão B consome mais memória do computador. Já a principal diferença entre as versões B e C é que a última abre 1024 processos, aumentando seu potencial ofensivo.

O **Sasser** possui muitas semelhanças com o **Blaster**, surgido em agosto de 2003, mas parece ultrapassar seu antecessor em tudo. Apenas 24 horas após ter sido identificado pela **Panda**[29], o **Sasser** já era encontrado em 3% dos computadores analisados pelo antivírus da empresa, contra 2,5% do **Blaster** no mesmo período de tempo. Ambos surgiram poucos dias após a divulgação de falhas de segurança pela Microsoft: 26 dias, no caso do **Blaster**, 18 dias no do **Sasser**. Além disso, as características técnicas também são semelhantes: afetam os sistemas **Windows** 2000 e XP reproduzindo-se automaticamente através de redes, varrem endereços **IP**[30] aleatórios para identificar alvos potenciais e utilizam o protocolo **FTP** para efetivamente infectarem a vítima. Os dois causam o reinício de máquinas atacadas que não

muitos servidores FTP estão disponíveis para uso público.

25. *Registry* ou Registro. Local onde são armazenadas informações essenciais para o funcionamento do Sistema Operacional Windows e outros aplicativos. Muitos vírus e outras ameaças utilizam o *registry* para serem iniciados automaticamente junto com o SO Windows.

26. *Mutex* é a sigla de ***mut****ual* ***ex****clusion object* e se refere a um programa que serve para controlar o acesso a recursos do sistema e evitar que mais de um processo utilize o mesmo recurso ao mesmo tempo.

27. Empresa voltada para segurança da informação.

28. Empresa voltada para segurança da informação.

29. Empresa voltada para segurança da informação.

30. "*Internet Protocol*", ou Protocolo Internet. O protocolo responsável pelo roteamento e entrega dos pacotes de informação é o protocolo IP, portanto o endereço utilizado neste processo é designado endereço IP. O endereço IP deve conter a identificação da rede ao qual o host pertence (endereço de rede) e o seu endereço dentro dessa rede (endereço de host). O endereço de um host numa rede TCP/IP é composto de 4 bytes(32 bits). A representação de um endereço IP pode ser: 172.21.5.2. O IP foi introduzido em 1978 e foi padronizado em 1981. O protocolo IP não confirma a recepção dos dados pelo computador de destino nem retransmite os dados em caso de erro, pois esse controle é feito pelo

protocolo TCP. O único controle realizado pelo IP é a eliminação de pacotes que passem por um número de roteadores maior que o especificado no campo TTL(*Time to Live*). Isso evita que pacotes com problema de roteamento circulem pela rede indefinidamente, gerando tráfego excessivo. A versão atual do IP é a versão 4 (IPv4), que não é alterada nem atualizada desde os anos 70. Após tanto tempo é natural que apresente alguns problemas em função dos progressos ocorridos na informática.

possuem as correções instaladas.

Rumores já apontam o **Sasser** como causador de grandes problemas em redes bancárias e de transporte aéreo e ferroviário em todo o mundo. A **Bloomberg**[31] informa que o banco Goldman & Sachs de Hong Kong foi infectado pelo ***worm*** e passa por dificuldades em sua rede. O **Herald Sun**[32] afirma que o terceiro maior banco finlandês também foi afetado e fechou suas 130 agências para evitar problemas mais graves. Outras duas notícias citam apenas "problemas com computadores" neste fim de semana, mas especula-se que a origem destes seja o **Sasser**.

O **Daily Telegraph**[33], reproduzido pela **News Interactive**[34], afirma que atrasos nos trens de Sidney, Austrália, que atingiram 300 mil passageiros, podem ter sido causados pelo **Sasser**. Notícia da Associated Press (AP) divulgada pelo **News 24 Hours Houston**[35] afirma que a **Delta Airlines**[36] cancelou 40 vôos e atrasou outros 32 em Atlanta, por problemas em seus computadores.

31. Empresa responsável por notícias e informações financeiras.

A **Microsoft** chegou a criar uma página de informações exclusivamente sobre o **Sasser**. Na página **"O que você deve saber sobre o worm Sasser e suas variações"** (em inglês), a empresa mantém uma ferramenta para identificar máquinas contaminadas e remover o ***worm***, além de fornecer informações sobre como corrigir e contornar a brecha de segurança que permite a ação do **Sasser** e *links* para as principais empresas antivírus. A **Panda** estima que a incrível quantidade de 300 milhões de computadores estejam vulneráveis em todo o mundo, e a taxa de infecção deve crescer.

32. Jornal australiano.

33. Jornal australiano.

34. Jornal australiano.

Empresas de antivírus divulgaram texto dos programadores do **Netsky**, encontrado em meio ao código da 29ª versão do ***worm***, a AC, que afirma serem eles os criadores do **Sasser**. A mensagem diz, em inglês:

35. Site de notícias americano.

36. Empresa de transporte aéreo.

*- "Ei, empresas av, vocês sabem que nós é que programamos o vírus sasser?*
*É, é verdade! Por que vocês o chamaram de sasser?*
*Uma Dica: Comparem o código do servidor FTP com o do Skynet-V!!!*
*LooL [gargalhadas]! Nós somos o Skynet..."*

## Prisão

Segundo informações disponibilizadas pelo portal **InfoGuerra**, um alemão de 18 anos preso no último fim de semana (08-09/05/2004), confessou ser o autor não só do **Sasser**, como se suspeitava desde o início, mas também de todas as versões do **Netsky**. De acordo com as autoridades que o interrogaram, o jovem **Sven Jaschan** deu detalhes sobre os vírus e sua disseminação.

Ele afirmou que a intenção era criar o **Netsky-A** para combater o **Mydoom** e o **Bagle**, apagando esses vírus dos computadores infectados. Na tentativa, surgiu o **Sasser**. No computador do acusado foi encontrado o código-fonte do **Sasser**, ponto determinante na investigação por sabotagem de computadores que o adolescente enfrentará. A pena para o crime pode chegar a cinco anos de prisão. Ele responde ao processo em liberdade e deverá sofrer uma punição abrandada, já que cometeu os crimes quando ainda tinha 17 anos. Vítimas

dos vírus ainda podem tentar obter indenizações por danos.

Segundo a agência **Reuters**, **Jaschan** podia estar tentando ajudar sua mãe, proprietária de uma pequena empresa de manutenção de computadores, a "**PC Help**", localizada na cidade de 920 habitantes onde vivem. A hipótese foi levantada pela revista **Der Spiegel** e não foi descartada pelos promotores responsáveis pelo caso. O pai e a madrasta de **Jaschan** afirmam que ele tentava apenas encontrar um antídoto e não tinha intenção de causar danos.

A **Microsoft** afirma ter recebido informações sobre o autor do vírus na quarta-feira. O trabalho de localização da origem do vírus contou com a ajuda do **FBI**, do serviço secreto norte-americano e da polícia alemã. O advogado da empresa disse que os informantes conheciam o jovem e não o descobriram pela análise técnica do vírus. Eles devem receber **US$ 250.000,00** pelas informações.

O simples fato de estar "*online*", ou conectado, às redes de computadores é um potencial risco para a segurança dos sistemas. Infinidades de tipos de dispositivos, protocolos e aplicações, compartilhando de um mesmo meio levam à tona inúmeras vulnerabilidades, tanto de software quanto de hardware, portanto, investimentos em segurança são necessários atualmente para manter a estabilidade da informação corporativa ou de usuários domésticos.

# 3. Proposta de Sistema Anti-Invasão

O estudo para um sistema anti-invasão foi voltado para as ameaças do tipo ***backdoor/trojan horse*** (porta dos fundos/cavalo-de-tróia), porém, algumas proteções aqui discutidas podem abranger também **vírus**, que incorporem certas funcionalidades.

A proposta apresentada não almeja ser uma única solução, ou uma solução definitiva para o tema "segurança da informação digital", nem tão pouco ser comparada à um sistema ***firewall*** ou sistema antivírus.

A abordagem utilizada supõe a existência de duas entidades externas que quando utilizadas em conjunto pelo ***hacker***, causam uma invasão computacional. Estas entidades externas são respectivamente: um **programa servidor (**tratado no decorrer como **ps)** e um **programa cliente(**tratado no decorrer como **pc)**. No mundo real, para exemplificar, o **ps** atua como um porteiro que permite a entrada de assaltantes, sendo considerado como um comparsa ou ajudante do invasor, enquanto que o assaltante pode ser considerado como o **pc**, ou seja, quem toma as decisões do que fazer, ou quais elementos e pertences, do ambiente ao qual se encontra, ele irá roubar / destruir / eliminar. É assumida também a existência de uma entidade externa de nome "Usuário", que será efetivamente a pessoa que operará o sistema e estará fornecendo informações e recebendo ***feedback***[37].

O **ps** pode ser também um **vírus** que tenha a funcionalidade de servidor. Como no exemplo do mundo real, no mundo digital o **ps** está infiltrado no ambiente do usuário que será invadido. Este programa pode se infiltrar em um ambiente computacional como um anexo de e-mail, anexo de comunicador instantâneo ou mesmo se escondendo embutido dentro de outros arquivos. O **pc** fica localizado no computador do ***hacker***, sendo que este o utiliza para se conectar ao **ps**. Se o **ps** não estiver em execução, o **pc** não consegue se conectar, não havendo possibilidade de invasão. Sem o **ps** o **pc** não funciona e este é um ponto importante da análise. Sem o **pc** o invasor também não consegue se conectar ao **ps**, porém, uma conexão a um **ps** pode ocorrer, mesmo que o ***hacker*** utilize outro **pc** que não o correto, mas de nada esta conexão vai adiantar pelo simples fato deste "novo **pc**" não conhecer o protocolo de comunicação do **ps** ao qual está conectado. O protocolo de comunicação entre o **ps** e o **pc** é muito importante pois é através dele que é feita a comunicação e troca de informação (execução de comandos arbitrários, *download* de arquivos, etc).

O **ps**, quando iniciado, executa uma série de procedimentos em seu processo de infecção, alguns possíveis procedimentos são listados abaixo, não necessariamente ocorrem na ordem apresentada:

1. - Se renomeia, para dificultar a detecção, geralmente este novo nome é randômico;
2. - Cria uma cópia de si mesmo, em algum diretório existente ou mesmo cria outros diretórios;
3. - Grava-se no registro do Windows ou em arquivos de inicialização para ser iniciado automaticamente com o Sistema Operacional;
4. - Abre uma **porta**, geralmente **TCP**, e fica pronto para receber conexões;

37. O termo "*feedback*" – traduzido como retroação, realimentação, retroalimentação e também como servomecanismo – é um mecanismo que serve para regular o funcionamento de um sistema através da comparação entre os resultados alcançados em relação ao padrão de resultados pré-estabelecidos. A retroação permite que o sistema seja controlado (regulado) para que as saídas reais sejam iguais (ou o mais próximo possível) as que foram planejadas. Para o funcionamento da retroação, uma parte da saída do sistema – geralmente informação – é utilizada para comparação com o que foi planejado, esperado, chamado de especificação, padrão pré-determinado, etc. É muito comum que este padrão tenha uma faixa de tolerância, dentro da qual o resultado seja considerado aceitável.

38. Sigla que significa "*User Datagram Protocol*". O protocolo UDP é um serviço de entrega de datagramas sem conexão e, portanto, não garante a entrega nem a integridade das informações. O computador de origem envia o datagrama e não pede confirmação do recebimento ao computador de destino. Ao contrário do TCP, o UDP: não se preocupa com a perda de mensagens; não retransmite datagramas faltantes; não ordena datagramas recebidos fora de ordem; não elimina datagramas duplicados; não reconhece o recebimento de datagrmas; não estabelece ou encerra conexões. O protocolo UDP é portanto um serviço não confiável, pois é suscetível à perda de informação. As rotinas de verificação de erros do UDP são mais simples do que o sofisticado controle de erros do protocolo TCP, o que torna o UDP mais rápido que o TCP. Por essa razão o UDP geralmente é utilizado na transmissão de sinais onde a velocidade de transmissão é mais importante que a confiabilidade dos dados, como sinais de voz e vídeo. Nesses casos, mesmo que uma quantidade de informação pequena seja perdida, a mensagem ainda continua inteligível. Num mecanismo de transmissão sem conexão, o receptor não sabe a prioridade que lhe será enviada uma mensagem. Se uma mensagem for perdida durante a transmissão, o receptor não saberá que deixou de receber

aquela mensagem. O emissor, por sua vez, não se preocupa em verificar se a mensagem enviada foi realmente recebida. Exemplos de aplicações que utilizam o UDP: *Network File System* (NFS), *Simple Network Management Protocol* (SNMP).

39. Estado do protocolo TCP para determinada porta, significando que a porta está aberta, aguardando solicitações de conexões. Os possíveis estados são: closed, listen, syn_rcvd, syn_sent, established, close_wait, fin_wait_1, closing, fin_wait_2, time_wait e last_ack.

5. - Avisa o(s) atacante(s) que a máquina está infectada;

6. - Coleta os endereços de e-mail salvos na máquina e se envia para estes endereços.

Após analisados as ações efetuados pelas ameaças digitais atualmente, o sistema anti-invasão proposto teria as seguintes funcionalidades:

**1. Monitorar o Registro(*Registry*) do Windows;**

O monitor do **registro** deveria permitir a inclusão, edição e exclusão de chaves (*keys*) para monitoramento. O monitor deveria permitir que o usuário fosse avisado de alterações ocorridas nas chaves em monitoramento. O usuário deveria poder cadastrar uma regra de ação para cada chave em monitoramento. A regra de ação seria acionada no momento em que uma modificação ocorresse em alguma chave monitorada do **registro**, i. e., seria uma ação, ou providência, tomada em decorrência de modificações.

Algumas regras de ação poderiam ser: **0-Nenhuma regra**, onde nenhuma regra seria aplicada quando houvesse alguma modificação na chave monitorada; **1-Apagar novos valores**, onde novos valores seriam apagados da chave monitorada no momento em que fossem criados; **2-Apagar novos valores e arquivos associados**, onde novos valores seriam apagados da chave monitorada no momento em que fossem criados e, também, se houverem arquivos associados a estas novas entradas, estes arquivos também seriam apagados de seus diretórios de origem; **3-Apagar novos valores e mover arquivos associados para a pasta de quarentena**, onde novos valores seriam apagados da chave monitorada no momento em que fossem criados e, também, se houverem arquivos associados a estas novas entradas, estes arquivos seriam movidos para a pasta de quarentena; **4-Apenas mover arquivos associados para a pasta de quarentena**, onde arquivos associados a novas entradas seriam movidos para a pasta de quarentena. Para toda ocorrência de modificação nas chaves monitoradas, o monitor do **registro** deveria ser capaz de gravar um log, para posterior análise ou providências por parte do usuário. O monitor do **registro** deveria ter uma opção de modo interativo, i.e., conforme ocorressem modificações em uma chave, o monitor do **registro** poderia exibir uma tela de aviso das modificações ocorridas e ações tomadas, ou apenas aplicar a regra mas sem exibir uma tela de aviso das modificações.

**2. Monitorar Portas de Conexão;**

O monitor de **portas** deveria ter a capacidade de monitorar as **portas** do protocolo **TCP** e do protocolo **UDP**[38]. Para tanto, o monitor de **portas** deveria ter a funcionalidade de incluir uma **porta** a monitorar, editar uma **porta** monitorada, e, excluir uma **porta** do monitoramento. No momento da inclusão de uma **porta** à ser monitorada, o usuário teria que escolher o número da **porta**, que seria de 0 à 65535, tanto para **TCP** quanto para **UDP**. O monitor de **portas** deveria ter a capacidade de cadastrar uma regra de ação para cada **porta** adicionada ao monitor. Se a **porta** monitorada for **TCP,** devido ao protocolo **TCP** ser orientado à conexão, a regra de ação seria acionada no momento em que uma tentativa de conexão fosse feita, i. e., seria uma ação, ou providência, tomada em decorrência de tentativas de conexão. Se a **porta** monitorada for **UDP**, devido ao protocolo **UDP** não ser orientado à conexão, a regra de ação seria acionada no momento em que fossem enviados dados para a **porta**, i. e., seria uma ação, ou providência, tomada em decorrência de dados enviados para a **porta** monitorada.

Algumas regras de ação para **portas TCP** seriam: **0-Bloquear acesso**, onde o acesso à **porta** seria bloqueado, tanto internamente, quanto externamente, i. e., um programa servidor (**ps**) que queira abrir esta **porta**, colocando-a em modo ***listenning***[39] para permitir que outros ***hosts*** tenham acesso à esta **porta**, não teria sucesso, e, também, programas cliente (**pc**) que tentem conectar-se à **porta** não obterão sucesso; **1-Bloquear acesso mas enviar mensagem**, onde o acesso à **porta** seria bloqueado, tanto internamente, quanto externamente, mas quando o pc tentar se conectar a **porta**, antes de ser derrubado, o monitor de **portas** irá lhe enviar uma mensagem. Esta mensagem poderia ser algo do tipo: *"Esta conexão não foi permitida e está sendo encerrada. Seu acesso foi negado aos recursos deste computador. As informações desta tentativa de conexão foram gravadas por motivos de segurança"*, ou alguma frase escolhida pelo usuário; **2-Bloquear acesso mas permitir interação com o protocolo do invasor**, onde o acesso à **porta** seria bloqueado, tanto internamente, quanto externamente, mas no momento da tentativa de conexão, uma tela de log da conexão apareceria, e iria mostrar os comandos que fossem enviados por parte do **pc**, mas também permitiria que o usuário enviasse comandos(*strings*) ao **pc** invasor.

Algumas regras de ação para **portas UDP** seriam: **0-Bloquear acesso**, onde o acesso à **porta** seria bloqueado, tanto internamente, quanto externamente; **1-Bloquear acesso mas enviar mensagem**, onde o acesso à **porta** seria bloqueado, tanto internamente, quanto externamente, mas quando o **pc** enviar dados a **porta**, o monitor de **portas** irá lhe enviar uma mensagem; **2-Bloquear acesso mas permitir interação com o protocolo do invasor**, onde o acesso à **porta** seria bloqueado, tanto internamente, quanto externamente, mas no momento em que fossem enviados dados à **porta**, uma tela de registro da conexão apareceria, e iria mostrar os comandos que fossem enviados por parte do **pc**, mas também permitiria que o usuário enviasse comandos(*strings*) ao **pc** invasor. O monitor de portas deveria ter a opção de modo interativo, i. e., conforme ocorressem tentativas de conexão no caso do **TCP** e envio de dados no caso do **UDP**, o monitor de portas poderia ou não exibir uma tela com as ocorrências. O monitor de portas deveria registras log de todas as ocorrências relacionadas às portas monitoradas, para posterior análise e providências por parte do usuário.

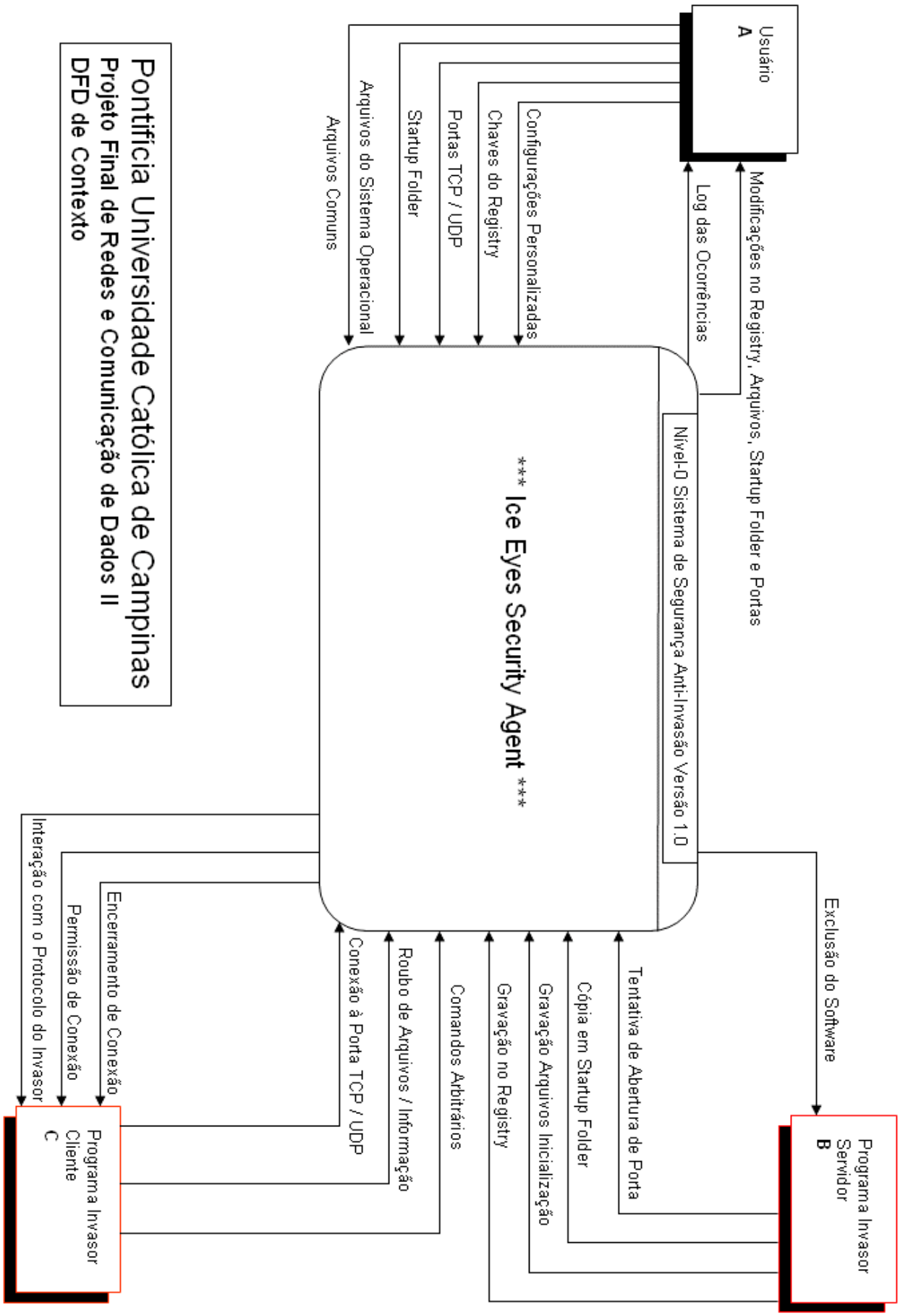
**3. Monitorar arquivos de inicialização do sistema operacional;**

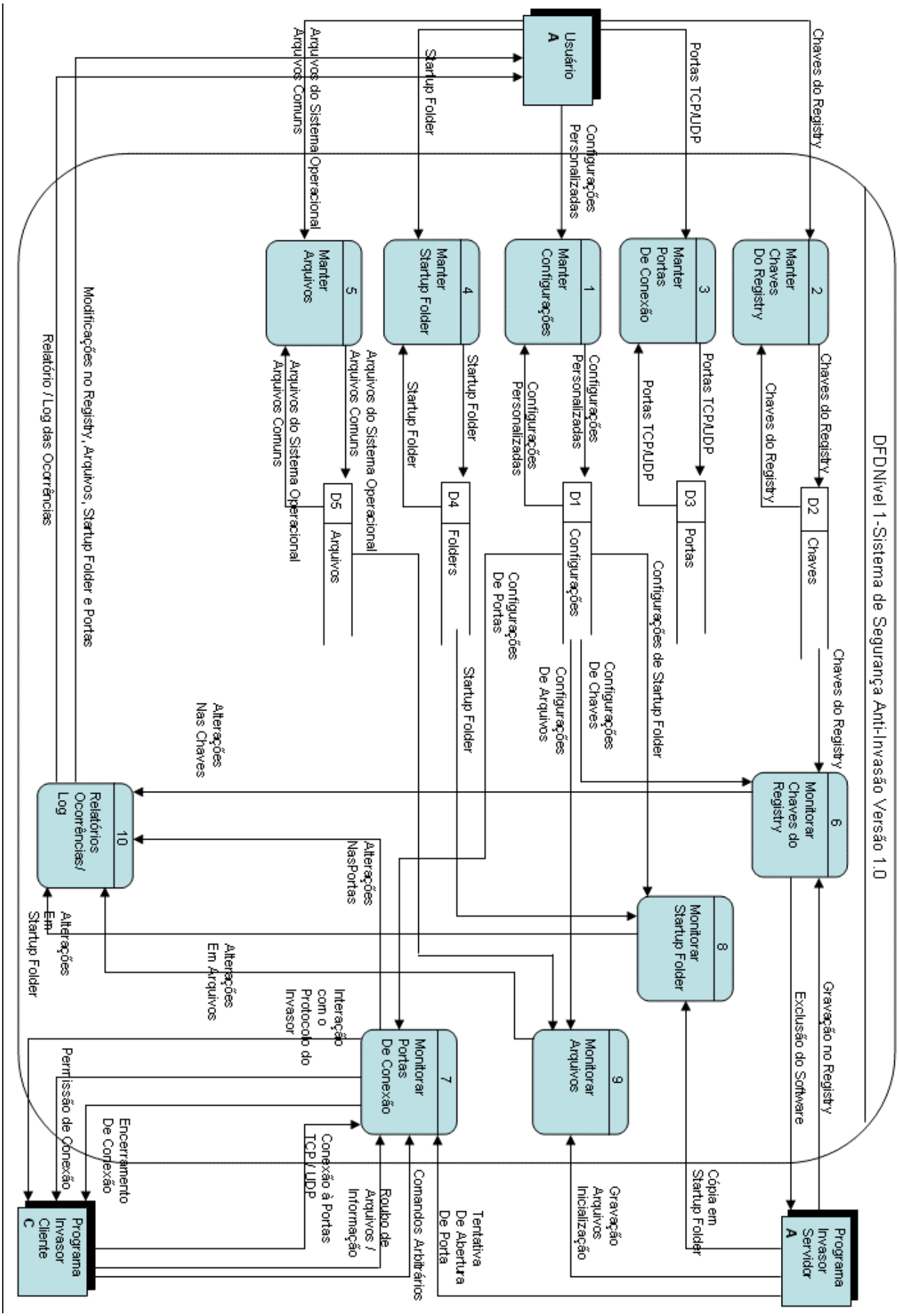
O monitor de arquivos de inicialização deveria permitir a inclusão, edição e exclusão de arquivos com conteúdo textual de qualquer extensão ao monitoramento. Se houverem modificações nestes arquivos o monitor de arquivos deveria apresentar uma tela exibindo as modificações ocorridas (o arquivo antigo e o arquivo modificado), para que o usuário possa fazer comparações e identificar as modificações. O monitor de arquivos deveria manter uma cópia dos arquivos que estão em monitoramento por motivos de segurança, caso algum problema ocorra com eles. O monitor de arquivos deveria fornecer a possibilidade de se cadastrar regras de ação para cada arquivo incluído ao monitoramento. Algumas regras de ação poderiam ser: **0-Nenhuma regra**, onde nenhuma regra de ação seria aplicada ao arquivo, as modificações apenas seriam exibidas e / ou registradas; **1-Restaurar arquivo**, onde as modificações seriam exibidas e / ou registradas, mas o arquivo seria restaurado com os dados antes da modificação; **2-Restaurar arquivo antigo, movendo novo arquivo para quarentena**, onde as modificações seriam exibidas e / ou registradas, mas o arquivo seria restaurado com os dados antes da modificação, porém, uma cópia do arquivo modificado seria guardada na pasta de quarentena, para posterior análise por parte do usuário. O monitor de arquivos deveria registrar log de todas as ocorrências relacionadas aos arquivos monitorados, para posterior análise e providências por parte do usuário.

**4. Visualizador de janelas ativas;**

A intenção do visualizador de janelas ativas seria exibir as janelas que estão atualmente carregadas na memória, mesmo as escondidas. O visualizador deveria ter a opção de mostrar(*show*) e esconder(*hide*) as janelas que estão atualmente carregadas na memória. O visualizador de janelas deveria ter o recurso de gerar um relatório com o título das janelas carregadas na memória.

# 4. Diagramas de Fluxo de Dados


Pontifícia Universidade Católica de Campinas
Projeto Final de Redes e Comunicação de Dados II
DFD de Contexto
Usuário
A
Log das Ocorrências
Modificações no Registry, Arquivos, Startup Folder e Portas
Configurações Personalizadas
Chaves do Registry
Portas TCP / UDP
Startup Folder
Arquivos do Sistema Operacional
Arquivos Comuns
*** Ice Eyes Security Agent ***
Nível-0 Sistema de Segurança Anti-Invasão Versão 1.0
Exclusão do Software
Programa Invasor
Servidor
B
Tentativa de Abertura de Porta
Cópia em Startup Folder
Gravação Arquivos Inicialização
Gravação no Registry
Comandos Arbitrários
Roubo de Arquivos / Informação
Conexão à Porta TCP / UDP
Encerramento de Conexão
Permissão de Conexão
Interação com o Protocolo do Invasor
Programa Invasor
Cliente
C


DFD Nível 1 - Sistema de Segurança Anti-Invasão Versão 1.0
Usuário A
Programa Invasor Servidor A
Programa Invasor Cliente C
1 Manter Configurações
2 Manter Chaves Do Registry
3 Manter Portas De Conexão
4 Manter Startup Folder
5 Manter Arquivos
6 Monitorar Chaves do Registry
7 Monitorar Portas De Conexão
8 Monitorar Startup Folder
9 Monitorar Arquivos
10 Relatórios Ocorrências/ Log
D1 Configurações
D2 Chaves
D3 Portas
D4 Folders
D5 Arquivos
Chaves do Registry
Portas TCP/UDP
Configurações Personalizadas
Startup Folder
Arquivos do Sistema Operacional
Arquivos Comuns
Modificações no Registry, Arquivos, Startup Folder e Portas
Relatório / Log das Ocorrências
Alterações Nas Chaves
Alterações NasPortas
Alterações Em Startup Folder
Alterações Em Arquivos
Interação com o Protocolo do Invasor
Permissão de Conexão
Encerramento De Conexão
Conexão à Portas TCP / UDP
Roubo de Arquivos / Informação
Comandos Arbitrários
Tentativa De Abertura De Porta
Gravação Arquivos Inicialização
Cópia em Startup Folder
Gravação no Registry
Exclusão do Software

# 5. Conclusão

Todos precisam ter consciência que os computadores quando interligados, são uma porta aberta para o mundo, com a agravante de não se poder ver quem o está olhando. Quem compartilha um universo tão diversificado, deveria, independentemente de qualquer coisa, prevenir-se contra surpresas desagradáveis.

A comparação mórbida da Teoria Geral de Sistemas é inevitável. Se o equilíbrio homeostático não resistir àquele fluxo de desintegração e corrupção, o ser humano começará a se desintegrar, mais do que pode se reconstruir, e assim, morrerá. Quando uma nova vulnerabilidade é descoberta e explorada, o equilíbrio homeostático é quebrado e a queda é vertiginosa para a **entropia** (caos).

Todos morreremos um dia, uns mais cedo e outros mais tarde. Esta variação de tempo de vida tem muitas influências como: qualidade de vida, localização de moradia, alimentação saudável, prática de esportes, uso de drogas, acidentes, etc. Os sistemas também irão parar de funcionar (*crash*), esta também é sua tendência natural, devido à influências do meio em que se encontra, má utilização do *software* (*peopleware* desinformada ou anti-ética), má construção do *software* (componentes internos, atualizações de versões). Porém isto pode ser prevenido com o auxílio de sistemas antivírus, sistemas *firewall* (anti-invasão), sistemas de *backup*, política de segurança, etc.

Não haverá nunca um sistema com falha zero, isto é fato, pela própria natureza humana, no mundo real, que até onde conhecemos atualmente não existe ser humano imortal, quanto pela natureza binária, no mundo digital. A “segurança da informação” é ponto chave para a estabilidade das empresas e continuidade de seus negócios, e desde os primórdios, onde nem se pensava em computação já existia o dito que “**prevenir é melhor que remediar**”, portanto, a prevenção contra ameaças digitais nunca é demais, mesmo quando se trata de usuários domésticos como grandes corporações internacionais.

# 6. Links

Módulo Security

http://www.modulo.com.br

Linux Security

http://www.linuxsecurity.com.br

SyHunt Corp.

http://www.syhunt.com

F-Secure

http://www.f-secure.com

Symantec Corp.

http://www.symantec.com

Sophos

http://www.linuxsecurity.com.br

Windows Update

http://windowsupdate.microsoft.com

Security Open Source

http://www.securityopensource.org.br

InfoGuerra – Segurança e Privacidade

http://www.infoguerra.com.br

Hackers To Hackers Conference

http://www.h2hc.com.br

Panda Software

http://www.pandasoftware.com

BR-Linux

http://www.br-linux.com.br

Unicamp Security Team

http://www.security.unicamp.br

SecForum

http://www.secforum.com.br

Security Focus

http://www.securityfocus.com

PUC Campinas

http://www.puc-campinas.edu.br

Takedown

http://www.takedown.com

The Ice-Eyes Project – Security against Digital Threats

http://ice-eyes.cjb.net

Security Focus

http://www.securityfocus.com

Net Security

http://www.net-security.org

Insecure Net

http://www.insecurenet.com.br

Packet Storm Security

http://packetstormsecurity.org

Net Security

http://www.net-security.org

# 7. Bibliografia

**[JOHN VON NEUMANN, 1949]** Neumann, John von. – "*Theory and Organization of Complicated Automata*". "*Edited and completed by Arthur W. Burks in 1966*". University of Illinois Press- Urbana and London., Illinois, USA, 1966.

**[G. B. BURN]** Burn, G. B. - "Por dentro do registro do Windows NT". São Paulo, SP. Editora Market Books do Brasil Ltda., 1999.

**[STEPHEN NORTHCUTT, 2000]** Northcutt, Stephen. – "Como detectar invasão em rede – um guia para analistas". Editora Ciência Moderna Ltda., Rio de Janeiro, 2000.

**[W. RICHARD STEVENS, 2000]** Stevens, W. Richard - "*TCP/IP Illustrated: the protocols*". Addison-Wesley Publishing Company, 2000.

**[LUCIANA PALMA E RUBENS PRATES, 2000]** Palma, Luciana – Prates, Rubens. - "TCP/IP – Guia de Consulta Rápida". Novatec Editora, 2000.

**[GUILHERME CESTAROLLI SELEGUIM E FELIPE BARBI, 2002]** Seleguim, Guilherme Cestarolli – Barbi, Felipe. – "Perigos do Mundo Virtual". Trabalho acadêmico sobre segurança computacional apresentado no curso de Análise de Sistemas – Pontifícia Universidade Católica de Campinas (PUCC), Campinas, Outubro / 2002.

**[GUILHERME CESTAROLLI SELEGUIM, 2002]** Seleguim, Guilherme Cestarolli. - “Cavalo de Tróia – Análise Histórica versus Análise Computacional”. Artigo sobre segurança computacional e cavalos de tróia publicado no portal Módulo Security. Novembro / 2002.

**[GUILHERME CESTAROLLI SELEGUIM, 2004]** Seleguim, Guilherme Cestarolli. - “Acesso Remoto”. Documento em português sobre cavalos de tróia publicado no Portal Linux Security. Fevereiro / 2004.

**[MÓDULO SECURITY, 2004]** Portal Módulo Security. – http://www.modulo.com.br – Acessado em: 21/05/2004.

**[INFOGUERRA, 2004]** Portal InfoGuerra – Segurança e Privacidade. – http://www.infoguerra.com.br – Acessado em: 21/05/2004.

**[LINUXSECURITY, 2004]** Portal Linux Security. – http://www.linuxsecurity.com.br – Acessado em: 21/05/2004.