

# Estudo de Mitnick e sua viabilidade nos dias atuais

Eduardo Martins Pereira (*eduardomp@globo.com*)

Fernando Bracalente (*bracalente@hotmail.com*)

Marcelo Dinofre (*mdinofre@hotmail.com*)

Mario Luiz Bernardinelli (*mariolb@gmail.com*)

25/Abril/2008

## Abstract

*The main purpose of this article is analyze the possibility of a new Mitnick's attack nowadays, including the analysis of all attack phases, its techniques, which vulnerabilities were explored and also if the actual operational system are susceptible to this kind of attack.*

## 1. Introdução

Este artigo visa analisar a possibilidade de um ataque de Mitnick nos dias atuais, analisando todas as fases do ataque, suas técnicas, quais foram às vulnerabilidades exploradas e se os sistemas computacionais atuais ainda estão suscetíveis a esse tipo de ataque.

## 2. Quem é Kevin Mitnick?



**Kevin Mitnick** foi um hacker americano, conhecido mundialmente a partir dos anos 90. Atualmente trabalha como consultor de uma empresa de segurança. Mitnick, que cometeu os primeiros delitos em 1980. Invadiu vários computadores,

como o da NSA, do centro de espionagem dos Estados Unidos, de operadora de celulares, de empresas de tecnologia e provedores de internet. Foi preso em 1995 e libertado em 2000 após pagar uma fiança de U\$4000. Ficou três anos em liberdade condicional, sem poder conectar-se à internet. Hoje trabalha como consultor de segurança na Web.

## 3. O ataque

O ataque de *Kevin Mitnick* contra os computadores de *Tsutomu Shimomura* ocorreu na noite de Natal de 1994. Este ataque utilizou três técnicas distintas:

- *IP spoofing*
- Sequestro de sessão
- Negação de serviço (*DoS*)

Conforme relato de Shimomura[1], três computadores de sua rede foram envolvidos no ataque:

- *target*. O objetivo final de Mitnick.
- *server*. Servidor que Mitnick personificou para poder ganhar acesso ao computador *x-terminal*.
- *x-terminal*. Estação de trabalho que Mitnick invadiu e que foi utilizada para acessar o objetivo final: o computador *target*.

O ataque de Mitnick partiu de um host comprometido na rede *toad.com*, que pertence a John Gilmore[6]. Deste computador comprometido, Mitnick executou os seguintes comandos:

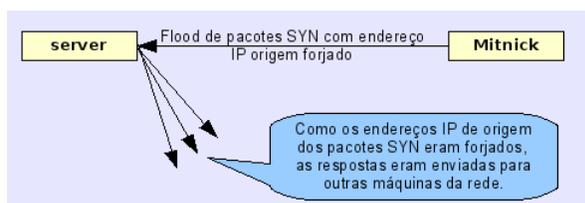
```
toad.com# finger -l @target
toad.com# finger -l @server
toad.com# finger -l root@server
toad.com# finger -l @x-terminal
toad.com# showmount -e x-terminal
toad.com# rpcinfo -p x-terminal
toad.com# finger -l root@x-terminal
```

Os comandos apresentados foram obtidos dos registros de *log* das máquinas de Shimomura [1].

Com estes comandos, Mitnick descobriu uma relação de confiança entre o computador chamado *server* e o *x-terminal*, que eram equipamentos SPARCStation que utilizavam o sistema operacional Solaris versão 1. Esta relação de confiança significa que o computador *server* podia acessar alguns recursos do computador *x-terminal* sem autenticação de usuário.

O segundo passo foi deixar o *server* indisponível (ataque *DoS*), de forma que ele não pudesse responder à nenhuma solicitação de *x-terminal*. A técnica utilizada para provocar *DoS* foi enviar uma enxurrada de pedidos

de início de conexão (pacotes *TCP SYN*) para a porta 513 de *server*, que é a porta padrão do serviço *rshell*. Com muitos pedidos de conexão, a fila de conexões ficou cheia e *server* ficou incapaz de responder à novos pedidos de conexão. Mitnick combinou este ataque com a técnica conhecida como *IP Spoofing*, na qual o endereço IP de origem dos pacotes são adulterados. Mitnick utilizou endereços IP não utilizados na rede, de forma a fazer com que, para cada pacote *TCP SYN/ACK* enviado por *server*, a resposta (*TCP ACK*) nunca retornasse, vencendo por *timeout*. Isto lhe garantiu mais tempo com a fila de conexões de *server* cheia. Veja o diagrama a seguir:



O terceiro passo foi descobrir o número de seqüência inicial a ser utilizado para atacar o computador de Shimomura. Mitnick fez isso usando um computador comprometido da rede *luc.edu* (Loyola University Chicago), conforme descrito em [7]. Mitnick enviou 20 pacotes *TCP SYN* deste computador para o computador *x-terminal*, de Shimomura. A cada pacote *TCP SYN/ACK* recebido de *xterminal*, Mitnick enviava um pacote *TCP RST*, de forma a finalizar a conexão e não levantar suspeitas nos registros de *log* de *x-terminal*. Este fato pôde ser evidenciado pelos registros de *log* publicados por Shimomura[1]:

```
apollo.it.luc.edu.1000 > x-terminal.shell: S
1382726990:1382726990(0) win 4096
x-terminal.shell > apollo.it.luc.edu.1000: S
2021824000:2021824000(0) ack 1382726991 win 4096
apollo.it.luc.edu.1000 > x-terminal.shell: R
1382726991:1382726991(0) win 0
apollo.it.luc.edu.999 > x-terminal.shell: S
1382726991:1382726991(0) win 4096
x-terminal.shell > apollo.it.luc.edu.999: S
2021952000:2021952000(0) ack 1382726992 win 4096
apollo.it.luc.edu.999 > x-terminal.shell: R
1382726992:1382726992(0) win 0
apollo.it.luc.edu.999 > x-terminal.shell: R
1382726992:1382726992(0) win 0
apollo.it.luc.edu.998 > x-terminal.shell: S
1382726992:1382726992(0) win 4096
x-terminal.shell > apollo.it.luc.edu.998: S
2022080000:2022080000(0) ack 1382726993 win 4096
```

Analisando os pacotes de resposta (*TCP SYN/ACK*), Mitnick descobriu que, a cada nova conexão o número de seqüência inicial era incrementado de 128000. No exemplo, a primeira requisição de conexão gerou o número de seqüência 2021824000, a segunda gerou o número 2021952000 e a terceira o número

2022080000, ou seja, uma diferença de 128000 para cada conexão.

Cabe aqui a observação de um fato que pode ter sido passado despercebido que é a data do ataque: Mitnick não escolheu a noite de Natal por acaso. Para que ele pudesse ter mais precisão na análise do comportamento da pilha TCP do computador alvo, era necessário que o mesmo estivesse com pouca ou nenhuma carga, isto é, sem solicitações de conexão.

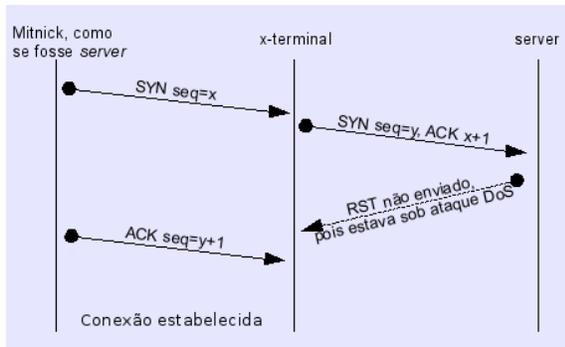
Agora Mitnick já tinha tudo o que precisava para desferir o ataque à *x-terminal*: já sabia como a pilha TCP de *x-terminal* gerava o número de seqüência inicial e o computador *server* estava incapaz de responder à novas conexões.

O objetivo de Mitnick era utilizar da relação de confiança entre *server* e *x-terminal* para forçar uma conexão à *x-terminal* como se fosse *server*.

Mitnick enviou então um pedido de conexão, isto é, um pacote *TCP SYN* para *x-terminal* com o endereço IP de *server* como endereço de origem.

O *x-terminal* respondeu então com um pacote *TCP SYN/ACK* para *server*. Numa situação destas e em condições normais, o computador *server* deveria responder ao *x-terminal* com um pacote *TCP RST*, pois ele não reconheceria o pacote *TCP SYN/ACK* recebido de *x-terminal*, já que ele não havia iniciado nenhuma conexão. No entanto, como *server* estava sob ataque, com a fila de conexões cheia, nenhuma resposta foi enviada por ele à *x-terminal*.

Por outro lado, como Mitnick conhecia o número de seqüência inicial, ele forjou novamente um pacote, agora *TCP ACK*, e enviou ao *x-terminal* como se fosse *server*. Desta forma o *three-way handshaking* necessário ao estabelecimento de conexões TCP estava concluído. Agora, Mitnick poderia enviar quaisquer comandos para *x-terminal*, já que podia manipular corretamente o número de seqüência e forjar as respostas de *server*. A figura a seguir ilustra o processo:



Observe que, mesmo tendo estabelecido a conexão, Mitnick ainda não receberia nenhuma resposta aos pacotes enviados, pois todas as respostas eram enviadas para *server*. A conexão explorada por Mitnick é de acesso remoto, conhecida por *rlogin* (ou *rshell*). Neste tipo de serviço, comandos podem ser executados de máquinas remotas sem a solicitação de senhas, ou seja, há uma relação de confiança entre as máquinas baseada no endereço IP das mesmas.

Uma vez estabelecida a conexão, Mitnick injetou um pacote na conexão contendo o comando:

```
echo + + >> /.rhosts
```

O arquivo *.rhosts* descreve as relações de confiança, isto é, quais computadores podem acessar livremente o sistema. O comando injetado por Mitnick adiciona ao final do arquivo *.rhosts* a seqüência *++*, que faz com que a máquina atacada passe a confiar em qualquer máquina. Agora, Mitnick poderia passar a acessar o computador *x-terminal* de qualquer computador.

Injetado o comando desejado, Mitnick injetou mais um pacote na conexão, agora *TCP FIN*, indicado à *x-terminal* que a conexão estava sendo finalizada.

Depois disso, Mitnick enviou pacotes *RST* para *server*, esvaziando a fila de conexões, fazendo com que ele voltasse a operar normalmente, finalizando assim o ataque DoS. Todo este processo demorou apenas 16 segundos, conforme informações obtidas no relato do ataque divulgado por Shimomura[1].

Com o acesso a *x-terminal* garantido, Mitnick pôde completar o ataque compilando e instalando *backdoors* em *x-terminal*: o caminho estava livre. Mitnick instalou um módulo de *software* conhecido como *tap-2.01*, que é um módulo de *kernel* que permite o seqüestro de sessões. Mitnick observou então as conexões entre *x-terminal* e outros servidores e, com o módulo instalado pôde tomar o controle de uma das sessões autenticadas com um dos servidores de Shimomura (*target*). O módulo *tap-2.01* permite que se tome o controle de um

terminal. Observe que, de acordo com o relato de Shimomura[1], Mitnick usou a invasão ao computador *x-terminal* para conseguir acesso a outro servidor (*target*).

#### 4. Como mitigar o ataque

O ataque realizado por Mitnick possui várias fases e, portanto, deve ser mitigado de várias formas.

A primeira delas é configurar o *firewall* de forma evitar que sejam obtidas informações a respeito do sistema a partir da Internet.

Outra atitude importante, mas nem sempre seguida, evite habilitar serviços desnecessários. Por exemplo, os serviços de acesso remoto, como *rlogin*, não devem ser disponibilizados para a Internet.

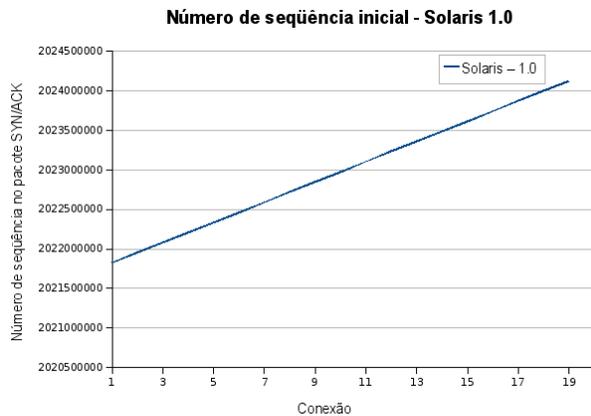
É recomendado manter atualizados os programas e migrar para novas ferramentas que proporcionam maior segurança como migrar o *Telnet* e o *rlogin* para o *SSH*, com isso os dados serão criptografados e se a sessão for monitorada será difícil visualizar as informações. Da mesma forma, o *FTP* pode ser substituído pelo *sFTP* minimizando a possibilidade do seqüestro de sessão. O uso *IPSec* possui características que previnem o seqüestro de sessão e, portanto deve ser utilizado sempre que possível.

#### 5. O ataque de Mitnick nos dias atuais

Qual a possibilidade de um ataque destes ter sucesso atualmente? A resposta para esta pergunta é difícil, pois depende de uma série de fatores.

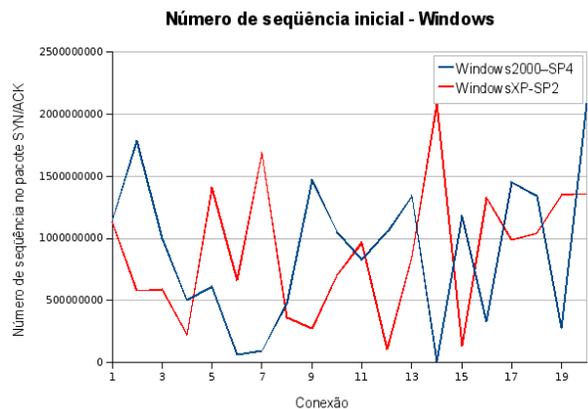
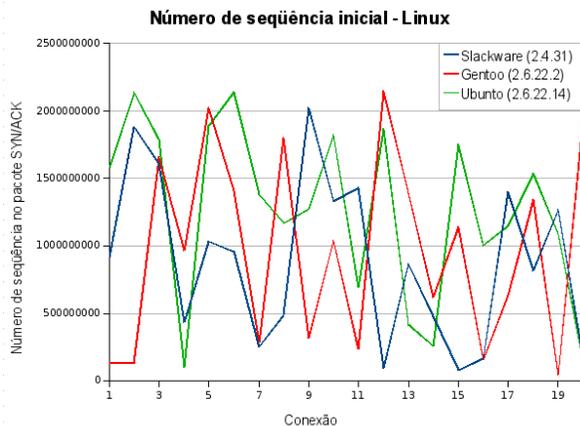
Em primeiro lugar, os serviços de acesso remoto (*rlogin* etc) estão caindo em desuso em favor de ferramentas mais seguras e eficientes, como por exemplo, o *SSH*. Portanto, a probabilidade de encontrar um servidor com serviços *rshell* disponíveis na Internet parece ser bastante baixa.

Outro fator importante é o número de seqüência inicial gerado pelo *TCP*. Usando as informações publicadas por Shimomura[1], se plotarmos num gráfico os números de seqüência obtidos por Mitnick, teremos o seguinte:



Observe que é facilmente previsível como o número de seqüência inicial era gerado naquela versão de sistema operacional.

Atualmente, determinar com precisão o número de seqüência inicial é uma tarefa mais difícil: os sistemas operacionais utilizam algoritmos mais eficientes para escolher o número de seqüência inicial de uma conexão. Nos diagramas abaixo estão representados os números de seqüência inicial obtidos em tentativas consecutivas de conexão a servidores com sistemas operacionais relativamente novos. Em todos os exemplos, foram enviados 20 pacotes TCP-SYN com destino a uma porta ativa qualquer no servidor e o número de seqüência devolvido pelo servidor foi anotado no gráfico. É importante observar que todos os sistemas estavam em completo repouso, isto é, as únicas conexões recebidas pelos servidores foram as utilizadas no teste.



## 6. Conclusão

Todos os ataques, até hoje são utilizados, pois exploram vulnerabilidades da arquitetura do protocolo TCP/IP, sendo assim são criados vários mecanismos de se contornar essas vulnerabilidades, é aí que surge o protocolo IP v6, que foi construído pensando em corrigir as falhas de segurança da versão 4.

## 7. Referências

- [1] SHIMOMURA, Tsutomu. Technical Details of the Attack Described by Markoff in NYT. <http://www.gulker.com/ra/hack/tsattack.html>. 12/Outubro/1997.
- [2] NAKAMURA, Emilio Tissato e GEUS, Paulo Lício. Segurança de Redes em Ambientes Cooperativos. 3a. edição. Editora Futura. 2003
- [3] [http://pt.wikipedia.org/wiki/Kevin\\_Mitnick](http://pt.wikipedia.org/wiki/Kevin_Mitnick), visitado em 15/04/2008
- [4] <http://ww.gtsav.gatech.edu/students/studentcenter/archive/news/kevin.mitnick.story.vert.jpg>, visitado em 15/04/2008
- [5] Gráficos, de autoria própria, criados em 10/04/2008
- [6] GILMORE, John, [www.toad.com](http://www.toad.com)
- [7] Whitaker, Andrew e Newman, Daniel P. Penetration Testing and Network Defense. *Kevin Mitnick's Session Hijack Attack*. Cisco Press. 2005.