



Pensando em Segurança

Por Glaudson Ocampos

Intruders Tiger Team Security

<http://www.intruders.org.br/>

Exemplos de Amadurecimento em Segurança

Há aproximadamente três anos atrás, poucas pessoas recomendariam o uso de alguns softwares que possuíam fama de serem inseguros. A lista englobava daemons de grande porte como Bind, Sendmail e Wu-ftpd, que reinaram durante a década de 90.

O histórico de falhas desses programas assustava grande parte da comunidade de segurança e de administradores de rede, afinal, quem gostaria de usar um programa que tem um histórico considerável de falhas?



Uma das primeiras falhas relatadas nos anais da segurança de informação, foi um erro presente nas primeiras versões do sendmail. Esse erro era conhecido como “Sendmail Debug”, por volta de 1988. Não é por menos que essa falha foi a primeira a ser numerada pela famosa lista de segurança e vulnerabilidades Bugtraq, conforme a referência abaixo atesta:

<http://www.securityfocus.com/bid/1>

A vulnerabilidade era bem simples de ser explorada, bastava um atacante se conectar no servidor de email, utilizando um programa como telnet, na porta tcp de número 25 do servidor alvo e enviar um pipe symbol (|) como endereço do destinatário do email, seguido de um comando que ele desejava que fosse executado no servidor.

Essa falha foi utilizada no primeiro worm usado na internet, o worm de Robert Morris, também no ano de 1988.

No decorrer dos anos, o sendmail continuou apresentando falhas de segurança bastante comprometedoras. Algumas delas foram muito usadas na Internet durante a década de 90, e a equipe de desenvolvedores do sendmail parecia estar perdida diante da quantidade de possíveis bugs e vulnerabilidades que o seu software poderia vir a apresentar.

Algumas alternativas de servidores de email passaram a ganhar espaço, dentre elas se destacaram o Qmail e o Postfix. Esses programas eram mais simples e possuíam menos recursos que o sendmail, no entanto, tinham como foco melhor segurança e menos complexibilidade.

A aceitação desses softwares obrigou os desenvolvedores do sendmail colocarem as mãos na massa e investirem em auditoria de código e maiores recursos de segurança. Com isso em mente, o sendmail vem apresentando menos problemas de segurança e tem caminhado pra se tornar um software robusto e seguro, voltando novamente a ser aceito em sistemas onde havia perdido credibilidade.

Um ponto positivo para o sendmail é que o histórico de falhas que permite execução de código remota presentes no sendmail remonta a 2003, cerca de três anos atrás. Em 2006, uma possível falha de race condition foi encontrada, mas nenhum exploit foi disponibilizado.

Pensando em Segurança

A mesma trajetória de problemas que vimos com o sendmail também pode ser vista com o Bind. O Bind ainda é o programa mais usado como servidor de DNS na internet, mas durante um certo tempo perdeu muito espaço para alternativas mais seguras como o djbdns.

O histórico do Bind é assustador! Algumas das falhas presentes no Bind foram usadas por atacantes para obter acesso a centenas de milhares de máquinas ao redor do mundo. Pelo menos duas grandes falhas do Bind merecem uma atenção especial.

Em novembro de 1999, uma falha conhecida como Bind NXT Overflow foi publicada e um exploit criado pelo grupo de hackers ADM Crew tornou manifesto os problemas que essa falha acarretaria. Tratava-se de uma condição de buffer overflow presente na validação de NXT Records que acarretava execução de comandos em uma máquina remota.

Essa falha foi massivamente explorada na internet, causando prejuízos incalculáveis.

Uma outra falha no Bind que merece a nossa atenção, foi uma falha conhecida como Bind TSIG. Tratava-se de uma condição de frame pointer overflow(atualmente a técnica é mais conhecida como off-by-one) que permitia alterar o fluxo de execução do programa para executar comandos desejados pelo atacante.

Essa falha foi inicialmente descoberta por Anthony Osborne e John McDonald, um renomado pesquisador de falhas. No entanto, um exploit público feito pelo brasileiro Gustavo Scotti da Axur, tornou manifesto no mais alto grau, o problema de segurança.

Um outro exemplo interessante de observar é o do WU-ftpD que teve seu primeiro choque ao ser publicada uma condição de buffer overflow que permitia execução remota de comandos

através do comando PWD do protocolo FTP.

Um exploit criado pelo proeminente hacker “duke” era de fácil execução e permitia que muitos sistemas fossem invadidos com relativa facilidade.

O histórico de falhas do WUftpD foi considerável. Problemas envolvendo mal uso de instruções, buffer overflow, heap overflow, malloc heap overflow e format strings. Podemos ver com isso que o WUftpD foi vítima de quase todas as técnicas mais usadas de exploração remota.

Em Agosto de 2000, uma técnica de exploração remota conhecida como Format strings bug foi massificada. Essa técnica foi utilizada com enorme sucesso contra sistemas executando WUftpD. Com uma quantidade gigantesca de servidores vitimados, muitos administradores de rede e analistas de segurança sugeriram alternativas ao uso da WU-ftpD.

Por volta de Julho de 2003, uma condição de off-by-one foi encontrada no WU-ftpD. Novamente, uma falha que já deveria ter sido auditada se mostrou presente nesse software, afetando a sua imagem.

O WUftpD não é mais o servidor de FTP mais utilizado em todo mundo. A credibilidade desse software caiu muito por conta dos inúmeros problemas de segurança que apresentou.

Esses exemplos servem para ilustrar a necessidade de se pensar em segurança quando se projeta um sistema. Foram citados apenas três grandes exemplos presentes no mundo Linux/Unix. A plataforma Microsoft têm sido criticada pela falta de segurança em seus produtos. É de conhecimento público que o IIS(Servidor web da Microsoft) perdeu muito terreno para o Apache por conta das sérias falhas de segurança que apresentou no passado.

Pensando em Segurança

O mesmo caso se aplica ao Internet Explorer, que a cada dia que passa vê sua hegemonia ir por água abaixo com o crescimento do concorrente, Firefox. Se por um lado existe a necessidade de se fazer programas com inúmeros recursos, isso não deve servir de desculpas para não se investir em segurança.

Os exemplos de projetos que tiveram a segurança projetada desde o princípio se mostraram bem eficazes e têm se mantido como excelentes opções. Os casos do Qmail, DJBDNS e Proftpd(o autor recomenda não usar FTP) ilustram muito bem essa questão. Muitos usuários desses programas se encontram satisfeitos por conta da segurança e confiabilidade que eles têm demonstrado ao passar dos anos.

Quando se pensa em segurança, não se deve negligenciar o fator histórico. Sistemas sem auditoria estão propensos a apresentarem maiores problemas. Essa máxima se aplica aos sistemas de código fechado, onde a filosofia do código fechado impede que excelentes pesquisadores possam encontrar falhas e alertar quanto aos problemas.

Se você é programador ou coordena uma equipe de desenvolvimento, lembre-se que se a segurança não for desde o início parte ativa no processo de criação, não tenha dúvidas que amanhã você terá problemas e seu software pode até mesmo cair em desuso por causa do pânico que uma falha pode ocasionar num usuário.

Maiores informações:

<http://www.securityfocus.com/bid/1>
<http://www.securityfocus.com/bid/788>
<http://www.securityfocus.com/bid/2302>
<http://www.securityfocus.com/bid/2242>
<http://www.securityfocus.com/bid/2496>
<http://www.securityfocus.com/bid/1387>
http://packetstorm.linuxsecurity.com/0102-exploits/tsl_bind.c
<http://adm.freelsd.net/ADM/exploits/t666.c>

A Internet é como uma selva. Você acha que vai encontrar um gatinho?



Intruders Tiger Team Security
<http://www.intruders.org.br/>
Preparando você contra os perigos da Internet!

Conheça o nosso firewall de aplicação WEB. Uma super-ferramenta capaz de bloquear a grande maioria dos ataques WEB. Visite-nos:

<http://www.security.org.br/webdefender.htm>

