MINISTÉRIO DA DEFESA
EXÉRCITO BRASILEIRO
DEPARTAMENTO DE CIÊNCIA E TECNOLOGIA
INSTITUTO MILITAR DE ENGENHARIA
(Real Academia de Artilharia Fortificação e Desenho - 1792)

RAFAEL ANASTÁCIO ALVES

MARINA SILVA CRUZ

DIEGO BRETAS DE ALVARENGA CARVALHO

ESTUDO DAS VULNERABILIDADES EXISTENTES NOS SERVIDORES WEB APACHE E IIS

Rio de Janeiro

2013

INSTITUTO MILITAR DE ENGENHARIA

RAFAEL ANASTÁCIO ALVES

MARINA SILVA CRUZ

DIEGO BRETAS DE ALVARENGA CARVALHO

ESTUDO DAS VULNERABILIDADES EXISTENTES NOS SERVIDORES WEB APACHE E IIS

Iniciação à Pesquisa apresentada ao Curso de Graduação em Engenharia da Computação do Instituto Militar de Engenharia.

Orientador: Anderson Fernandes P. dos Santos, D.Sc.

Rio de Janeiro

2013

INSTITUTO MILITAR DE ENGENHARIA

DIEGO BRETAS DE ALVARENGA CARVALHO

MARINA SILVA CRUZ

RAFAEL ANASTÁCIO ALVES

ESTUDO DAS VULNERABILIDADES EXISTENTES NOS SERVIDORES WEB APACHE E IIS

Projeto de Iniciação à Pesquisa do Curso de Engenharia de Computação do Instituto Militar de Engenharia.

Orientador: Major Anderson Fernandes P. dos Santos

Aprovada em 13 de Junho de 2013 pela seguinte Banca Examinadora:

_____

Anderson Fernandes P. dos Santos - D.Sc. do IME.

_____

Raquel Coelho Gomes Pinto - D.Sc do IME.

_____

Julio Cesar Duarte - D.C. do IME.

Rio de Janeiro

2013

SUMÁRIO

RESUMO

Com o aumento e ampla abrangência da internet cresce a necessidade de garantir certo nível de segurança das informações transmitidas. Pichações de portal, juntamente com outros tipos de ataques a servidores Web apresentam uma ameaça para usuários. Elas estão diretamente ligadas ao tema do presente trabalho: o estudo das vulnerabilidades existentes nos servidores Web Apache e IIS.

O objetivo é estudar e classificar as vulnerabilidades existentes em servidores Web, de modo que essa classificação sirva de base para trabalhos futuros de outros grupos. Sendo assim, inicialmente foram estudados os servidores Web de um modo abrangente, visando suas características, funcionamento e distribuição no mercado. Tendo em vista que os servidores Web mais presentes atualmente são o Apache e IIS, foi então realizado um estudo específico de cada um, identificando suas vulnerabilidades. Com essa identificação, foram propostas nove categorias que englobam todas as vulnerabilidades observadas, sendo então possível catalogá-las.

Logo, próximas pesquisas podem contar com uma divisão das vulnerabilidades dos servidores Web Apache, versões 2.0, 2.2 e 2.4, e IIS em categorias que podem, por exemplo, ajudar sistemas a detectar certas vulnerabilidades e avisar aos usuários antes que elas ocorram ou, para futuramente, propor soluções inteligentes para tais vulnerabilidades.

ABSTRACT

With the increase and wide range of internet grows the need to ensure a certain level of security of information transmitted. Defacements along with other types of Web server attacks might represent a threat to users. They are directly related to the subject of this work, which has the purpose of studying the existents vulnerabilities at Apache and IIS Web servers.

The goal is to study and classify vulnerabilities in Web servers, so this classification became a basis for future work of other groups. Therefore, initially was studied the Web servers in a comprehensive way, aiming characteristics, operation and market share. Given that most Web servers currently used are Apache and IIS, then was made a specific study of each one, identifying their vulnerabilities. With this identification, were proposed nine categories that encompass all vulnerabilities observed, being then possible to catalog them.

Therefore, future researches may rely on a division of vulnerabilities in Apache Web servers, versions 2.0, 2.2 and 2.4, and IIS into categories that can, for example, help systems to detect certain vulnerabilities and warn users before they occur, or to future, propose smart solutions for such vulnerabilities.

1 INTRODUÇÃO

Atualmente os servidores web possuem um importante papel no que se diz respeito ao compartilhamento de dados e informações através da internet e o bom funcionamento desses sistemas, tanto a nível de software, quanto a nível de hardware, garante o máximo de desempenho e lucratividade das organizações que dependem do seu funcionamento.

Muitos avanços foram realizados em relação às características de hardware desses servidores, porém existe um vasto campo a ser explorado em torno das características de software. Dessa forma, os softwares possuem vulnerabilidades, isto é, estão sujeitos a sofrerem invasões que comprometem sua confidencialidade, disponibilidade ou sua integridade.

Essas vulnerabilidades são um ramo de destaque no estudo dos servidores web, visto que, a interrupção de algum serviço por si só já pode ser bastante prejudicial. Além disso, diversos servidores possuem informações confidenciais de pessoas, empresas ou até mesmo governos, e o comprometimento da segurança do servidor Web poderá acarretar no comprometimento da segurança dessas informações.

Para que se possa especificar e detalhar as vulnerabilidades de um servidor Web e a partir de uma análise dessas vulnerabilidades seja possível criar métodos mais eficientes para evitar que elas aconteçam, antes de tudo é necessário conhecimento de alguns elementos essenciais para o seu funcionamento, dentre eles: servidor Web, protocolo HTTP e protocolo TCP/IP.

1.1 MOTIVAÇÃO

Visto que a falta de segurança é um problema recorrente na internet atualmente e a própria falta de segurança de alguns servidores Web contribui para tal fato, a principal motivação deste trabalho é, a partir desse estudo, é fornecer uma base para estudos relacionados à segurança da Internet e das informações que nela são disponibilizadas.

Assim como outros estudos realizados no Instituto Militar de Engenharia, como, por exemplo, trabalhos na área de criptografia, este trabalho preza pela segurança da informação, algo essencial não só para o Exército Brasileiro, mas como para o governo como um todo. É importante ressaltar que o próprio Exército Brasileiro também utiliza os servidores web e caso algum de seus servidores seja invadido, pode acontecer de informações sigilosas serem comprometidas ou de algum serviço importante ser interrompido.

1.2 OBJETIVO

Este trabalho tem como objetivo realizar uma pesquisa sobre as vulnerabilidades dos servidores Web Apache e IIS, categorizando-as para que seja possível determinar quais tipos são mais recorrentes em cada servidor específico

1.3 JUSTIFICATIVA

Os servidores Web possuem uma função vital na internet atualmente, uma vez que, eles são essenciais para o funcionamento das páginas da Web e disponibilização de seus conteúdos. Entretanto, nos dias de hoje, é possível perceber que os ataques aos servidores ou a essas páginas têm sido frequentes, fato esse que pode comprometer não só algum tipo de serviço oferecido através dos servidores, como a segurança de informações confidenciais.

Para que seja possível reduzir esses ataques é necessário entender melhor as vulnerabilidades de cada tipo de servidor, então esse trabalho estudará os servidores Web Apache e IIS, visto que existe uma predominância muito grande desses servidores no mercado (NEFTCRAFT, 2013), fato que os torna mais importantes quando comparados aos outros servidores existentes atualmente. Além disso, será feita uma classificação dos tipos de vulnerabilidades desses servidores para que seja possível obter uma ferramenta para mitigar esses ataques.

1.4 METODOLOGIA

Este trabalho está estruturado em três etapas principais: a primeira delas se refere ao estudo dos servidores Web e compreensão de seu funcionamento, assim como dos elementos que foram considerados essenciais para o entendimento de como funciona um servidor Web.

A segunda etapa compreende o estudo dos servidores Web Apache e IIS e suas particularidades, assim como suas vulnerabilidades. A última etapa abrange a classificação e catalogação dessas vulnerabilidades para que sejam identificados os tipos de vulnerabilidades mais recorrentes em cada um dos servidores estudados.

1.5 ORGANIZAÇÃO

O capítulo 2 trata dos servidores Web em geral, explicando melhor o seu funcionamento, além disso, trata também de alguns elementos que são considerados essenciais para o entendimento do seu funcionamento, como, por exemplo, TCP/IP, HTTP.

Os capítulos 3 e 4 tratam, respectivamente, dos servidores Web Apache e IIS, da particularidade de cada um dos servidores.

Os capítulo 5 trata especificamente do assunto vulnerabilidades nos servidores em questão. Nele é realizada a classificação das vulnerabilidades de cada um dos servidores. E o capítulo 6 se refere à conclusão do trabalho.

2 SERVIDORES WEB

2.1 DEFINIÇÃO

Um servidor Web tem como principal função oferecer páginas Web como resposta aos pedidos de clientes utilizando o *Hypertext TransferProtocol* (HTTP). Entretanto, além de páginas Web ele também é responsável por fornecer diversos outros tipos de dados, como, por exemplo, vídeos em tempo real e transmissão de e-mails.

Um agente de usuário, normalmente um navegador inicia a comunicação fazendo um requerimento por um recurso específico usando HTTP e o servidor responde com o conteúdo ou uma mensagem de erro, em caso de indisponibilidade. O recurso tipicamente é um arquivo real armazenado na memória secundária, mas isso não é necessariamente sempre assim, depende de como o servidor Web é configurado.

Embora a principal função seja a de servir conteúdo, uma implementação completa de HTTP também inclui formas de recebimento de conteúdo dos clientes. Este recurso é utilizado para a apresentação de formulários Web, incluindo o *upload* de arquivos.

2.2 CARACTERÍSTICAS GERAIS

Cada servidor Web possui características específicas, porém, existe um conjunto de recursos presente na maioria deles. Como os servidores são construídos especificamente para hospedar sites, suas características são, muitas vezes, centradas em torno de criação e manutenção de ambiente de hospedagem de um site. Dentre essas características comuns, é possível citar:

- Especificar documentos padrão, isto é especificar documentos que são exibidos quando nenhum nome de arquivo for especificado;
- Manipular conteúdo estático e dinâmico;
- Criar diretórios virtuais e mapeá-los para diretórios físicos;
- Configurar quais contas de usuário possuem permissão para ver o site ou quais endereços de IP possuem permissão;

- Criar um site FTP. Um site FTP permite aos usuários transferir arquivos de e para o local;
- Criar um ou mais sites;
- Configurar páginas de erro personalizadas;
- Oferecer suporte ao HTTPS;
- *Virtual Hosting*- para servir muitos sites usando um endereço IP;
- Oferecer suporte a arquivos grandes.

2.3 ORIGENS

A origem dos servidores Web se deu em 1989, quando Tim Berners-Lee propôs um novo projeto para a gestão de informações enquanto trabalhava no CERN (*Conseil Européen por La Recherche Nucléaire*, atualmente conhecida como Organização Européia para a Pesquisa Nuclear). Esse projeto tinha como objetivo facilitar a troca de conhecimento entre cientistas desse instituto. O funcionamento baseava-se em ligar hipertexto entre computadores. Dessa forma, havia uma rede única de dados para ajudar os físicos do CERN a compartilhar todas as informações contidas em seus computadores.

Um dos resultados foi a criação de um navegador-editor com a intenção de elaborar um padrão para tornar a Web um espaço para compartilhar e editar as informações e construir um hipertexto comum.

2.4 CONTEÚDO

A forma mais simples de exibição do conteúdo na Web é o conteúdo estático, no qual os arquivos ficam armazenados no servidor para que possam ser fornecidos ao cliente que o requisita.

Entretanto, na maioria das vezes é utilizado o conteúdo dinâmico, pois em outros cenários de utilização de servidores Web, o usuário não necessariamente está requisitando um arquivo. Por exemplo, quando um usuário conectado a um servidor que disponibiliza uma página Web preenche o formulário contido nessa página Web e clica

em um botão com função *submit*, ele não está solicitando ao servidor que retorne algum conteúdo, mas, sim que processe os dados contidos nesse formulário (TANENBAUM, 2003).

Um dos mecanismos para processar formulários e outras páginas da Web dinâmicas é o do padrão CGI (*Comum Gateway Interface*). Ele permite aos servidores Web interagirem com programas e scripts de *back end* de forma que processem os dados de formulários ou outros conteúdos dinâmicos e gerem uma página HTML como resposta.

Outro procedimento utilizado para processamento e geração de conteúdo dinâmico é implementar scripts de pequena proporção dentro do HTML de forma a serem executados pelo servidor. A linguagem mais popular para geração desses scripts é o PHP (*Hypertext Preprocessor*).

Para que os scripts funcionem de fato, é necessário, porém, que o servidor seja capaz de identificar um módulo PHP para Apache. Para isso, essa linguagem foi desenvolvida para correto funcionamento em servidores Apache. Esse é um dos poucos servidores que permite a utilização de módulos que permite a sua configuração de forma a otimizá-lo para determinada demanda de conteúdo.

Outras tecnologias similares ao PHP são a JSP (*JavaServer Pages*), da Sun e a ASP (*Active Server Pages*), da Microsoft. Por terem padrões de linguagem e funcionalidade parecidos, a escolha da linguagem é uma questão política de optar por linguagem de procedências distintas, como código aberto, Sun ou Microsoft. (TANENBAUM,2003).

2.5 WEB SERVER FARMS

É um conjunto de computadores com funcionalidade de servidor de forma a suprir uma alta demanda em que normalmente somente uma máquina seria insuficiente. Todos esses computadores são conectados entre si numa rede por meio de *switches* ou roteadores que permitem a comunicação entre as diferentes partes do cluster e os respectivos usuários. Frequentemente os Web *server farms* contam com a existência de servidores de backup, que assumem a função em caso de falha dos servidores primários.

Os Web *server farms* são usados no contexto de *cluster computing*, ou seja, vários computadores são conectados entre si e configurados de forma a ter o comportamento de um único sistema. Através do método de balanceamento de carga alguns servidores distribuem as solicitações entre os demais, de forma que eles recebam um pequeno número de solicitações, se comparado a sua capacidade.

2.6 FUNCIONAMENTO

A comunicação com um servidor Web ocorre através de uma sequência de ações: um usuário, através de um navegador Web, inicia a comunicação fazendo um pedido para um recurso específico usando HTTP e o servidor responde com o conteúdo do recurso ou uma mensagem de erro, no caso de não ter sido possível fazê-lo. Essa é a função básica de um servidor Web.

Além dessa principal função, que é a de fornecer conteúdo, uma implementação completa de HTTP também possibilita formas de recebimento de conteúdo dos clientes. Esse tipo de recurso é utilizado para fazer, por exemplo, o upload de arquivos.

Os servidores Web nem sempre são utilizados na *WorldWide Web*. Muitas vezes eles podem ser usados incorporados em dispositivos como *webcams* e *routers* e que servem apenas para uma rede local. O servidor Web pode, portanto, ser usado como uma parte de um sistema para a administração de dispositivos como os citados acima.

Os servidores podem ser implementados no espaço do usuário ou no espaço de kernel. Quando instalado dentro do kernel, normalmente funcionarão mais rápido, já que, como ele faz parte do sistema, ele pode usar diretamente todos os recursos de hardware, tais como memória não paginada e adaptadores de redes.

Os servidores que são executados no espaço do usuário necessitam pedir ao sistema permissão para utilizar mais memória ou recursos da CPU. Não só essas requisições tomam tempo, como nem sempre são satisfeitas, uma vez que o sistema reserva recursos para seu uso próprio e para uso de outros aplicativos em execução.

2.7 DISTRIBUIÇÃO DE MERCADO DOS SERVIDORES WEB

Na figura 2.1 abaixo é possível ver a variação da distribuição de mercado dos servidores Web nos últimos 17 anos.

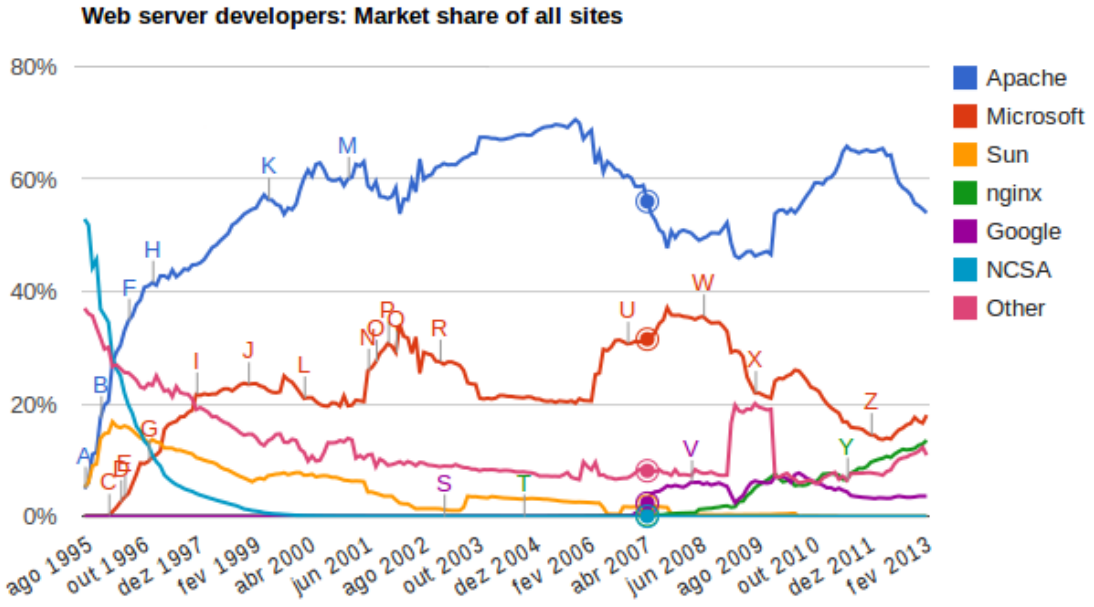

FIG 2.1 - Parcela de mercado de cada servidor Web entre 1995 e 2013.

(NEFTCRAFT, 2013)

Deste gráfico, observa-se a predominância de uso dos servidores Apache e IIS no mercado. É importante ressaltar também que nem sempre um servidor Web é utilizado para servir a Internet global, ou seja, existem também servidores incorporados a impressoras, *routers*, *webcams* e que servem apenas a uma rede local.

2.8 TCP/IP

O TCP/IP é uma pilha de protocolos de comunicação utilizados para a Internet e redes semelhantes, ele especifica como os dados devem ser formatados, endereçados, transmitidos, emitidos e recebidos no destino, e, portanto, dessa forma permite uma troca de informações entre dois computadores.

A arquitetura TCP/IP é dividida em quatro camadas, sendo elas:

- A camada de aplicação, responsável por oferecer serviços ao usuário;

- A camada de transporte, responsável pela transferência dos dados entre os dois sistemas finais;
- A camada de rede, responsável pelo empacotamento de dados em datagramas IP que contêm informações utilizadas para encaminhar datagramas entre hosts e redes. Executa também o roteamento de datagramas IP;
- A camada de acesso à rede, que tem a função de fazer com que as informações sejam transmitidas de um computador para o outro. É responsável também pela tradução do endereço físico para o endereço lógico.

Em relação à primeira camada, vale ressaltar que, ao desenvolver uma aplicação, é necessário escolher uma arquitetura de aplicação de rede. Para os servidores Web é utilizada a arquitetura cliente-servidor. Nesse modelo há sempre um host em funcionamento para atender as requisições dos clientes, esse host é denominado servidor. É importante ressaltar que nessa arquitetura os clientes não se comunicam entre si, uma vez que a requisição é feita ao servidor.

Para a camada de transporte, o protocolo utilizado pelos servidores Web é o TCP, esse protocolo oferece entrega de fluxo confiável, é orientado à conexão, mas se comparado ao UDP, o primeiro possui uma implementação mais complexa. Na prática, o TCP é o responsável por oferecer a confiabilidade que os usuários desejam em uma transmissão de dados. (COMER, 2006).

As principais características do TCP são: orientação à conexão, conexão *full-duplex*, confiabilidade, controle de fluxo, conexão ponto a ponto, transmissão em buffer e fluxo não-estruturado. (KUROSE, 2011)

A figura 2.2 mostra a estrutura do segmento TCP.

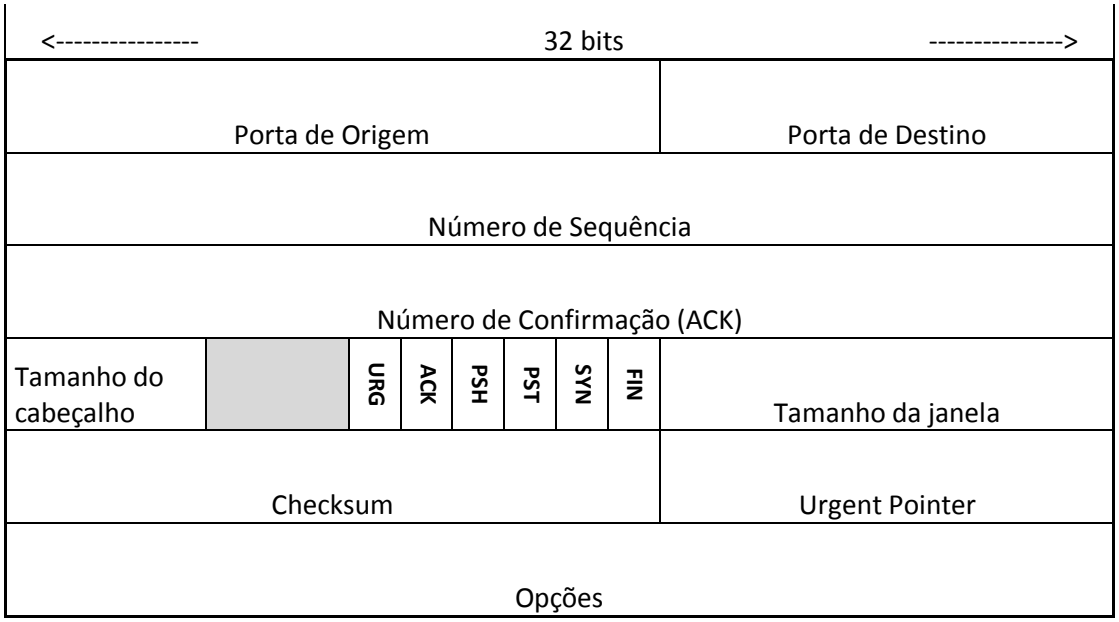

FIG 2.2 – Estrutura do segmento TCP.
(POSTEL, 1981)

O Protocolo da Internet, IP, é o utilizado pelos servidores Web na camada de rede, atualmente, a versão mais utilizada é o IPv4, mas já existe a proposta de substituição para o IPv6.

Uma das maiores motivações para se desenvolver o IPv6 foi a escassez de endereços de IP. Sendo assim, além de um espaço para endereços de IP maior, outras mudanças julgadas necessárias com base na experiência operacional também foram efetuadas. As principais mudanças que ocorreram no formato do datagrama foram; capacidade de endereçamento expandida de 32 bits para 128 bits, cabeçalho aprimorado de 40 bytes, rotulação de fluxo e prioridade, além da não possibilidade de fragmentação do datagrama.

Sabe-se que um pacote da camada de rede é denominado datagrama, mas ele pode ser diferente de acordo com a versão do protocolo. No IPv4, a estrutura desse datagrama é a disposta na figura 2.3.

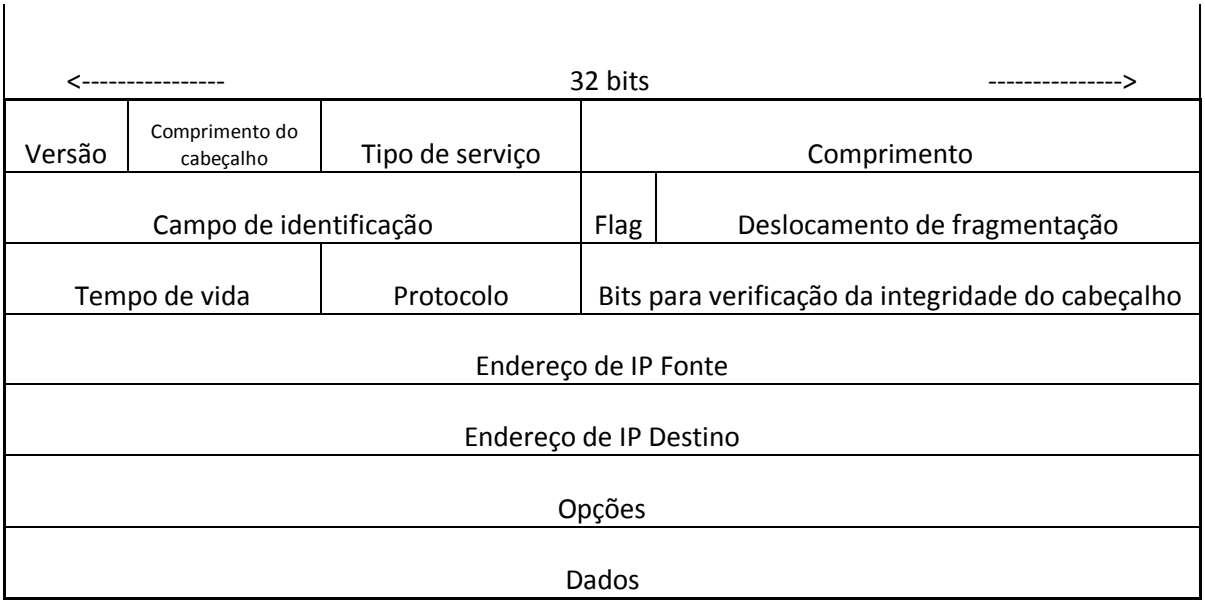

FIG 2.3 – Estrutura do datagrama IPv4.
(KOZIEROK,2005)

A partir desses campos que são determinadas as especificações do datagrama, como, por exemplo, a versão do protocolo, o comprimento do cabeçalho, o tipo de serviço, o endereço IP da fonte e do destino.

Vale ressaltar que quando o protocolo IPv4 foi criado não havia grande preocupação em relação à segurança das informações, uma vez que o estabelecimento da conexão por si só já era um grande desafio. Entretanto, com o passar dos anos a segurança dos dados se tornou algo primordial e por isso novos protocolos de camada de rede tiveram que ser desenvolvidos para auxiliar nesse fato.

O IPsec é um dos mais utilizados atualmente para auxiliar na segurança, desenvolvido para ser compatível tanto com o IPv4 quanto com o IPv6 (KUROSE, 2011).

A última camada, a de acesso à rede, não está muito clara na definição do modelo TCP/IP, mas em geral pode-se dizer que, desde que seja permitido o envio de pacotes IP, pode ser qualquer tipo de protocolo. Em geral, tem a função de fazer com que as informações sejam transmitidas de um computador para o outro. O protocolo dessa camada define o formato dos pacotes que serão trocados entre os nós nas extremidades do enlace, assim como as ações realizadas por esses nós ao enviar e receber os pacotes.

2.9 PROTOCOLO HTTP

O HTTP – Protocolo de Transferência de Hipertexto – é um protocolo da camada de aplicação. Ele surgiu da necessidade de se padronizar a comunicação entre cliente e servidor visando distribuir informações entre eles. O HTTP definiu a estrutura das mensagens trocadas entre cliente e servidor e o modo como eles as trocam. Sendo assim, este protocolo definiu como clientes Web requisitam páginas aos servidores e como eles as transferem a clientes.

O HTTP é implementado em dois programas: um programa cliente e outro servidor. Os dois programas, executados em sistemas finais diferentes, conversam um com o outro por meio da troca de mensagens HTTP. Quando um usuário requisita uma página Web, o navegador envia ao servidor mensagens de requisição HTTP para os objetos da página. O servidor, por sua vez, recebe as requisições e responde com mensagens de resposta HTTP que contém os objetos.

O HTTP utiliza conexões persistentes em seu modo padrão, mas os clientes e servidores HTTP podem ser configurados para utilizar a conexão não persistente.

Em conexões não persistentes cada conexão TCP transporta exatamente uma mensagem de requisição e uma mensagem de resposta e ela é encerrada logo após o servidor enviar o objeto. Sendo assim, no caso em que o cliente deseja obter informações de uma página Web que contém ‘n’ objetos, todos residentes no mesmo servidor, são geradas ‘n’ conexões TCP.

Por outro lado, em conexões persistentes, o servidor deixa a conexão TCP aberta após enviar resposta. Requisições e respostas subsequentes entre os mesmos clientes e servidor podem ser enviadas por meio da mesma conexão. No caso do exemplo anterior, os ‘n’ objetos poderiam ser obtidos através de apenas uma conexão TCP persistente.

Existem duas versões do protocolo HTTP, o HTTP/1.0 e o HTTP/1.1.

O HTTP/1.0 é um protocolo sem estado, seu funcionamento é como em uma conexão não persistente. Já o protocolo HTTP/1.1 tem configurada uma conexão persistente como padrão.

Na conexão persistente, é possível que uma conexão seja estabelecida para enviar várias requisições em sequência sem a necessidade de esperar por cada resposta. Elas serão recebidas na mesma ordem em que as solicitações foram enviadas. Esse processo é

chamado de *pipelining*. Pode também haver uma conexão sem *pipelining*, em que o cliente só faz nova requisição quando o servidor lhe envia a resposta, ou seja, cliente fica inativo até o objeto atingir o seu destino.

Um cache Web ou servidor Proxy é uma entidade da rede que possui seu próprio disco de armazenagem. Seu objetivo é agilizar o acesso a dados que foram recentemente requisitados pelo usuário. O navegador de um usuário pode ser configurado para que todas as suas solicitações passem primeiramente pelo cache Web.

Para um navegador requisitar um objeto, primeiramente, ele estabelece uma conexão TCP com o cache Web e envia a ele uma requisição HTTP para o objeto. O cache Web verifica se possui uma cópia do objeto armazenada localmente, em caso positivo ele envia uma cópia ao navegador do cliente dentro de uma mensagem HTTP. Caso negativo, o cache Web abre uma conexão TCP com o servidor de origem enviando uma requisição HTTP do objeto. Quando o servidor de origem recebe a requisição ele envia o objeto ao cache Web em uma mensagem de resposta HTTP. Ao receber a resposta, o cache Web armazena uma cópia em sua memória e envia o objeto solicitado ao navegador do cliente pela conexão TCP existente entre o navegador do cliente e o cache Web.

3 APACHE

3.1 CARACTERÍSTICAS

Várias são as características que tornaram o Apache tão popular:

- É configurável e utiliza módulos: existem módulos que são desenvolvidos por meio de linguagem C ou Perl que permite que o Apache atenda a demandas específicas de cada usuário.
- É gratuito e Open Source;
- Apache é otimizado para Perl, PHP e outras linguagens script: Perl, por exemplo, é a linguagem de script mais popular e dá suporte ao CGI e ao *mod_perl* (KABIR, 2002).
- Apache funciona em Linux e outros sistemas baseados em Unix: isso ajudou a tornar o servidor popular, pois o Linux é um sistema operacional bastante difundido entre os usuários que necessitem de um servidor Web. Outras sistemas Unix, tais como o FreeBSD, Solaris e Mac OS também ajudaram a aumentar o leque de usuários do Apache.
- Apache funciona também em Windows: Já existia versão compatível desde a versão 1.3.x do Apache, embora a 2.x tenha mais melhorias. Isso incentiva os usuários do servidor IIS a migrarem para o Apache.

3.2 RECURSOS

3.2.1.1 MÓDULOS DE MULTIPROCESSAMENTO

Antes dos módulos multiprocessamento, ou MPM (*Mutiprocessing Modules*) serem disponibilizados a partir da versão 2.0, o Apache utilizava a arquitetura *preforking*. Nessa arquitetura, os processos pai se subdividem em processos filhos, que ficam responsáveis em responder às requisições do momento. Os processos pai monitoravam os processos filho e os geravam ou encerravam baseados na quantidade de requisições recebidas. Essa arquitetura, porém, não funcionava adequadamente para as plataformas como o Windows, que, por não ser baseado em sistemas UNIX, e não utilizar o padrão POSIX (*Portable Operating System Interface*) para que houvesse compatibilidade.

Cada MPM é responsável por receber as requisições e responde-las por meio de processos filhos ou *threads*, dependendo de cada MPM utilizado. A seguir serão citadas e explicadas algumas delas.

3.2.1.2 MPM prefork

Baseada na arquitetura anterior à versão 2.0 do Apache, ela cria uma *pool* de processos filhos para responder às requisições. Cada processo filho tem uma e somente uma *thread*.

3.2.1.3 MPM threaded

Este MPM tornou possível a utilização de threads nas versões do Apache a partir da versão 2.0. Utiliza o mesmo conceito do prefork, mas com a adição da possibilidade de utilizar múltiplas threads em cada processo filho. Cada thread dentro de um processo filho pode responder a diferentes requisições.

3.2.1.4 MPM perchild

Este MPM permite que certo número de processos filho sejam inicializados com um número de threads especificado. À medida que o número de requisições aumenta os processos incluem novas threads conforme a necessidade. Quando a contagem de requisições diminui, os processos reduzem o número de threads.

A principal diferença, comparado com o MPM threaded, é que a perchild utiliza um número estático de processos e cada um deles podem ser acionados por diferentes usuários. Isso facilita a utilização de diferentes sites virtuais Web por vários usuários e grupos ID.

3.2.1.5 MPM winnt

Este MPM, do tipo *multithreaded*, foi criado para a plataforma Windows para substituir o uso do padrão POSIX para compatibilidade com Apache.

Ao utilizar esse módulo, o Apache irá criar um processo pai e um processo filho. O processo filho criará todos os threads que um determinado serviço requeira. Esse módulo também tem a vantagem da utilização de funções nativas ao Windows, o que

representa vantagem de desempenho em relação ao uso do padrão POSIX em versões anteriores ao Apache 2.x

3.2.1 FILTERING I/O

Esse recurso permite que os dados de saída resultantes do processamento de determinado módulo possam ser utilizados como dados de entrada para outros módulos.

Exemplo disso são os dados de saída do módulo CGI (mod_*cgi*), que podem conter tags referentes ao módulo SSI, e, assim, serem aproveitadas como dado de entrada para processamento sob esse módulo (KABIR, 2002).

3.2.2 CGI DAEMON

O módulo mod_cgi não funciona de maneira otimizada em MPMs que se baseiam em threads, porque elas não processam corretamente scripts CGI. Para MPMs baseados em threads, o módulo mod_cgid foi adicionado a partir da versão 2.0.

Dessa forma os scripts CGI passarão a ser executados da seguinte maneira (KABIR, 2002):

1. Quando a requisição por CGI chega a uma thread dentro do processo filho, ele a repassa para o CGI daemon;
2. O CGI daemon delega o script CGI e os dados de saída desse script e os coloca numa thread no processo filho;
3. A thread retorna os dados para o navegador Web.

3.2.3 APACHE PORTABLE RUN-TIME

Corresponde a um novo conceito introduzido nas versões a partir da 2.0. Antes disso, a portabilidade do software para vários sistemas operacionais era administrada de maneira interna a organização desenvolvedora do Apache de forma que se tornava trabalhoso o manuseio das versões do Apache de cada sistema. Com a introdução dessa novidade, a organização desenvolve somente uma única versão de código do Apache e criam-se várias bibliotecas a parte, nas quais será possível fornecer uma simples interface em C contendo funções específicas de cada sistema operacional. Dessa forma, essas bibliotecas tornam o Apache, agora disponível em uma única versão, compatível com todos os sistemas operacionais.

4 IIS

4.1 Características gerais

O IIS é o servidor Web criado pela Microsoft e que, diferentemente do Apache, possui compatibilidade apenas com os sistemas operacionais Windows. Atualmente, é o segundo servidor Web mais utilizado, perdendo apenas para o Apache (NEFTCRAFT, 2013). Outra diferença entre esses dois servidores é que o código do IIS não é aberto.

Em sua versão 7.5, o IIS apresenta suporte para diversos protocolos como HTTP, HTTPS, SMTP, NNTP, FTP e FTPS. Além de diferentes tipos de autenticação como, por exemplo: anônima, básica, UNC, entre outras.

4.2 Arquitetura

O IIS trabalha com uma arquitetura baseada em módulos, ou seja, ao invés de ter todas as funcionalidades instaladas de uma só vez, ele possui uma ferramenta na qual é possível adicionar ou remover esses componentes, chamados de módulos, de acordo com a sua necessidade. As principais vantagens que podem ser citadas dessa nova arquitetura são: a redução da área de ataque, uma vez que todos os componentes não utilizados podem ser removidos, além do aumento do desempenho, pelo mesmo motivo citado acima.

Esses módulos existentes são divididos em grupos de acordo com suas funcionalidades e alguns deles vem como padrão no IIS, esses grupos são:

- Módulos HTTP que incluem módulos para responder mensagens de erro, redirecionar requisições, filtrar requisições, entre outros. Como exemplos, é possível citar o *CustomErrorModule* que é responsável por enviar mensagens HTTP de erro;
- Módulo de segurança que são responsáveis por tarefas relacionadas à segurança dos processos realizados pelo servidor. Como exemplo, existe o *AnonymousAuthenticationModule* que é responsável pelas autenticações

anônimas e o *IpRestrictionModule* que restringe o acesso à apenas IPs presentes na lista ip*Security*.

- Módulos de conteúdo, esse grupo inclui métodos responsáveis por ações como listar o conteúdo de um diretório e servir páginas estáticas. Como exemplo, existe o Cgi*Module* que executa o *Common Gateway Interface* (CGI) e o *StaticFileModule* para gerar páginas estáticas.
- Módulos de compactação, nesse grupo existem apenas dois módulos, o *DynamicCompressionModule*, responsável por compactar respostas e o *StaticCompressionModule* responsável por pré-compactar conteúdos estáticos.
- Módulos de cache que desempenham tarefas relacionadas ao cache de respostas, como exemplo, pode-se citar o *UriCacheModule* que provê cache da informação URL em modo usuário e *HttpCacheModule* que provê cache no Http.sys tanto em modo usuário quanto em modo kernel.
- Módulos de *log* e diagnóstico são responsáveis por ações relacionadas à informações de diagnósticos e ao *log* de ações, ou seja, ao registro de ações. Dentre os módulos dessa categoria, pode-se citar o *CustomLoggingModule* e o *TracingModule* responsável por reportar eventos ao Microsoft *Event Tracing*.
- Módulos de suporte, responsáveis por proverem suporte aos módulos gerenciados, um exemplo é o *ConfigurationValidationModule* que é responsável por validar questões de configurações.
- Módulos de gerenciamento, todos as categorias citadas acima são consideradas nativas, uma vez que já vem juntamente com o IIS em seu estado inicial, já os módulos de gerenciamento são justamente aqueles que devem ser adicionados separadamente pelo o usuário, sendo que nessa categoria estão presentes módulos que desempenham as mais diversas funções. Como exemplo, é possível citar o *OutputCache* que suporta o cache de saída e *FormsAuthentication* que suporta autenticação de formulário.

4.3 Os protocolos de escuta (*Listeners Protocol*)

Os protocolos de escuta desempenham um trabalho essencial no funcionamento do ISS. Esse componente recebe uma requisição específica, a envia para o IIS para processamento e então envia a resposta de volta à quem a requisitou. Por padrão, o IIS utiliza o HTTP.sys como ouvinte para requisições HTTP e HTTPS.

4.4 Funcionamento

Dependendo da versão do IIS o gerenciamento dos processos, monitoração de desempenho e administração e configuração HTTP podem ser feito por diferentes componentes, até a versão 7.0 o *World Wide Web Publishing Service* (WWW *service*) era, sozinho, o responsável por essas atribuições, mas a partir dessa versão, o Windows *Process Activation Service* (WAS) começou a também ser utilizado para essas funções.

A figura 4.1, de acordo com o site oficial do IIS (www.iis.net), como os serviços e protocolos citados acima processam uma requisição HTTP:

1. O cliente faz uma requisição HTTP para o servidor Web. HTTP.sys intercepta a requisição;
2. HTTP.sys contata o WAS;
3. WAS requere informações sobre configuração da aplicação applicationHost.config;
4. O serviço WWW recebe as informações de configuração, como, por exemplo, configuração do site e pool de aplicação;
5. O serviço WWW utiliza essas informações para configurar o HTTP.sys;
6. WAS inicia um processo de trabalho para o pool de aplicação do qual a requisição foi feita;
7. O processo de trabalho processa a requisição e retorna uma resposta para o HTTP.sys;
8. O cliente recebe uma resposta.

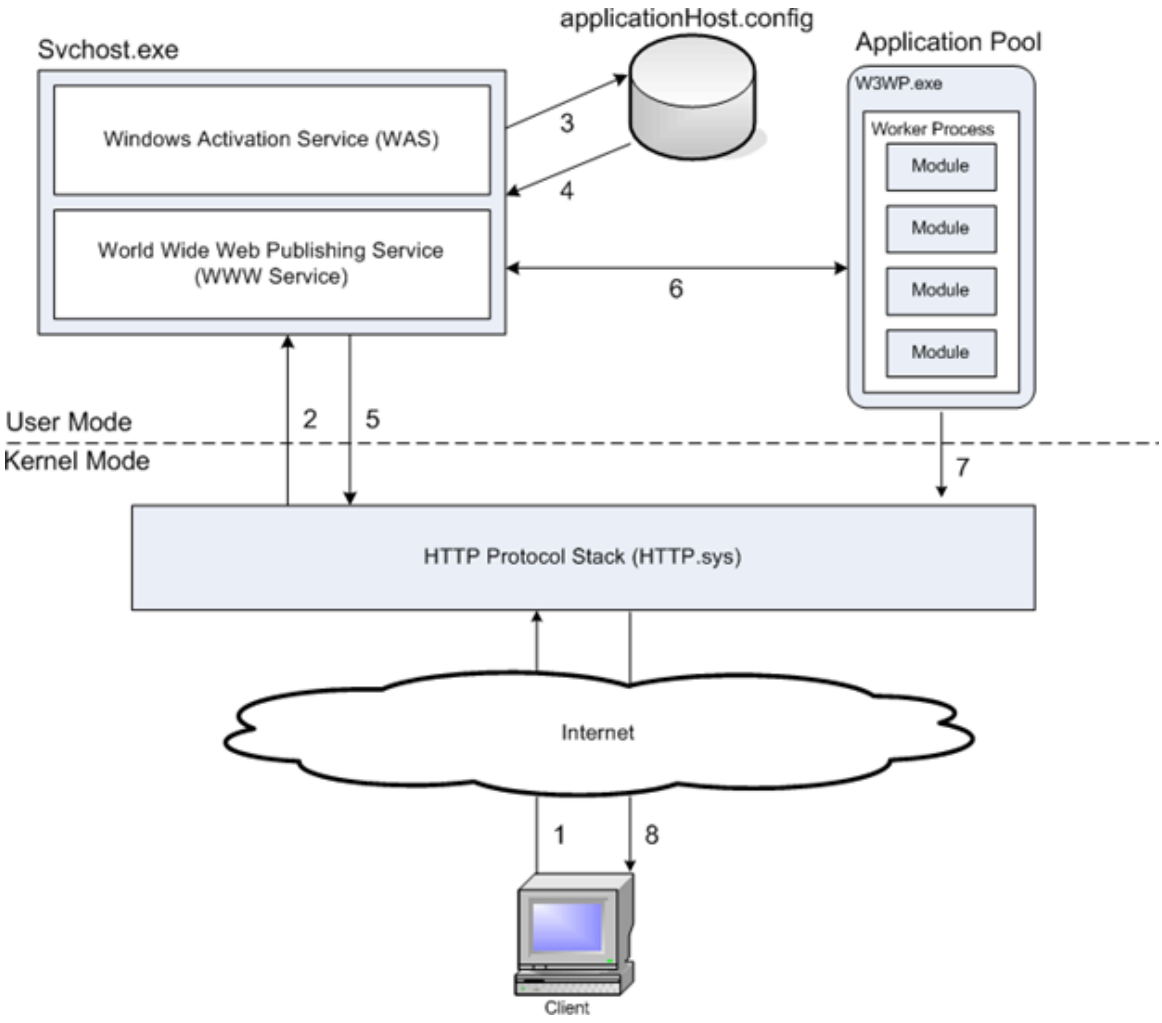

FIG 4.1 – Visão geral do processamento de uma requisição HTTP.

(IIS, 2013)

5 VULNERABILIDADE

5.1 INTRODUÇÃO

Neste trabalho será adotada a definição de vulnerabilidade referente à segurança. Segundo TECHNET, vulnerabilidade de segurança é: “uma fraqueza num produto que poderia permitir a um invasor comprometer a integridade, disponibilidade ou a confidencialidade daquele produto.”

Esse é um conceito que é utilizado pela *Microsoft Security Response Center* (MSRC) para categorizar os problemas de seus produtos e que fornece uma definição satisfatória para o estudo das vulnerabilidades dos servidores Web, seja a IIS, da própria Microsoft, seja o Apache, que é *open-source*.

Nos próximos itens serão abordados os conceitos de integridade, disponibilidade e confidencialidade.

- Integridade

  A integridade de um software está comprometida quando um invasor o modifica de maneira silenciosa e sem a devida autorização.

  No caso de servidores Web, esse tipo de invasão é categorizado como *defacement*. Exemplo disso ocorre quando sites têm sua aparência modificada por invasores, devido a objetivos diversos, tais como desafio pessoal ou protesto político.

- Disponibilidade

  Refere-se à capacidade de se acessar recursos de um software. Logo, quando um invasor limita o usuário de acessar determinados recursos, está limitando sua disponibilidade.

  No caso de servidores Web, isso ocorre quando ele faz com que determinado site fique fora do ar.

- Confidencialidade

  Se refere a capacidade de limitar o acesso a recursos e informações a usuários autorizados. Logo, quando o invasor obtém acesso a recursos ou informações restritas geridas por um software, está comprometendo a confidencialidade desse produto.

  Invasões de sites com o intuito de coletarem informações, é um exemplo de quebra de confidencialidade.

5.2 CRITÉRIO DE CLASSIFICAÇÃO

Este trabalho classificará as vulnerabilidades encontradas nos servidores Apache e ISS sob o seguinte processo:

Primeiro, classificará sob 9 grupos de vulnerabilidades: DoS (*Denial of Service*), execução de código, *overflow*, corrupção de memória, XSS (*Cross Site Scripting*), *Directory Traversal*, *by-pass*, obtenção de informação e obtenção de privilégio. Em seguida será feito um ordenamento desses grupos baseado na quantidade de registros para cada servidor.

A seguir, uma breve descrição de cada grupo.

- Negação de serviço

  Impede os usuários de utilizarem determinado serviço do servidor que virá a ser atacado (RISTIC, 2005). Isso é feito por meio do consumo dos recursos que são utilizados para o proveio desse serviço.

CVE-ID	
CVE-2011-3348	Learn more at National Vulnerability Database (NVD) • Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings
Description	
The mod_proxy_ajp module in the Apache HTTP Server before 2.2.21, when used with mod_proxy_balancer in certain configurations, allows remote attackers to cause a denial of service (temporary "error state" in the backend server) via a malformed HTTP request.	

Figura 5.1 – Exemplo de vulnerabilidade do tipo Negação de serviço.

- Execução de código

Corresponde a uma vulnerabilidade onde é executado código (script) malicioso no servidor web (RISTIC, 2005).

CVE-ID	
CVE-2011-3348	Learn more at National Vulnerability Database (NVD) • Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings
Description	
The mod_proxy_ajp module in the Apache HTTP Server before 2.2.21, when used with mod_proxy_balancer in certain configurations, allows remote attackers to cause a denial of service (temporary "error state" in the backend server) via a malformed HTTP request.	

Figura 5.2 – Exemplo de vulnerabilidade do tipo Execução de Código.

- *Overflow*

Corresponde a vulnerabilidades em que se ultrapassam limites para determinados tipos de dados armazenados no servidor web (RISTIC, 2005).

CVE-ID	
CVE-2003-0542	Learn more at National Vulnerability Database (NVD) • Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings
Description	
Multiple stack-based buffer overflows in (1) mod_alias and (2) mod_rewrite for Apache before 1.3.29 allow attackers to create configuration files to cause a denial of service (crash) or execute arbitrary code via a regular expression with more than 9 captures.	

Figura 5.3 – Exemplo de vulnerabilidade do tipo Overflow.

- Corrupção de memória

Ocorre quando o conteúdo de uma localização de memória é modificada. Quando algum destes conteúdos é posteriormente utilizado, ele gera um comportamento anormal ou até mesmo a interrupção do funcionamento do programa.

CVE-ID	
CVE-2003-0132	Learn more at National Vulnerability Database (NVD) • Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings
Description	
A memory leak in Apache 2.0 through 2.0.44 allows remote attackers to cause a denial of service (memory consumption) via large chunks of linefeed characters, which causes Apache to allocate 80 bytes for each linefeed.	

Figura 5.4 – Exemplo de vulnerabilidade do tipo Corrupção de Memória.

- XSS (Cross Site Scripting)

  Corresponde a uma vulnerabilidade em que se controla o código fonte HTML de uma página e se emite marcação de HTML e JavaScript a vontade. (RISTIC 2005).

CVE-ID	
CVE-2002-1593	Learn more at National Vulnerability Database (NVD) • Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings
Description	
mod_dav in Apache before 2.0.42 does not properly handle versioning hooks, which may allow remote attackers to kill a child process via a null dereference and cause a denial of service (CPU consumption) in a preforked multi-processing module.	

Figura 5.5 – Exemplo de vulnerabilidade do tipo XSS.

- *Directory Traversal*

  Corresponde a vulnerabilidade em que se utiliza *backreference* de diretório (../) em caminhos de arquivos para obter acesso a pastas acima da pasta indicada. (RISTIC, 2005). Se o servidor não consegue processar corretamente esses *backreferences*, ele também não consegue detectar o acesso aos arquivos vulneráveis a esse tipo de ataque. Essa vulnerabilidade pode existir no servidor web ou no código de aplicação que é utilizada nesse servidor.

CVE-ID	
CVE-2002-1744	Learn more at National Vulnerability Database (NVD) • Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings
Description	
Directory traversal vulnerability in CodeBrws.asp in Microsoft IIS 5.0 allows remote attackers to view source code and determine the existence of arbitrary files via a hex-encoded "%c0%ae%c0%ae" string, which is the Unicode representation for ".." (dot dot).	

Figura 5.6 – Exemplo de vulnerabilidade do tipo Directory Traversal.

- *By-pass*

  Se consegue burlar as restrições de usuário e de segurança do servidor web.

CVE-ID	
CVE-2003-0017	Learn more at National Vulnerability Database (NVD) • Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings
Description	
Apache 2.0 before 2.0.44 on Windows platforms allows remote attackers to obtain certain files via an HTTP request that ends in certain illegal characters such as ">", which causes a different filename to be processed and served.	

Figura 5.7 – Exemplo de vulnerabilidade do tipo By-pass.

- Obtenção de Informação

  Revela informações internas ao sistema, tais como diretório de código fonte e de outros arquivos restritos.

CVE-ID	
CVE-2004-0811	Learn more at National Vulnerability Database (NVD) • Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings
Description	
Unknown vulnerability in Apache 2.0.51 prevents "the merging of the Satisfy directive," which could allow attackers to obtain access to restricted resources contrary to the specified authentication configuration.	
References	

Figura 5.8 – Exemplo de vulnerabilidade do tipo Obtenção de Informação.

- Obtenção de Privilégio

Ganho de privilégio ocorre quando algum usuário ganha um acesso a recursos que normalmente ele não teria direito, geralmente esse acesso é permitido a apenas alguns tipos de usuários que necessitem ser autenticados anteriormente ou que seria disponível apenas ao administrador do sistema.

CVE-ID	
CVE-2004-0747	Learn more at National Vulnerability Database (NVD) • Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings
Description	
Buffer overflow in Apache 2.0.50 and earlier allows local users to gain apache privileges via a .htaccess file that causes the overflow during expansion of environment variables.	

Figura 5.9 – Exemplo de vulnerabilidade do tipo Obtenção de Privilégio.

Vale ressaltar, que uma mesma vulnerabilidade pode ser englobada em mais de uma categoria.

5.3 APACHE

Até o mês de março de 2013, a W3Techs divulgou a seguinte distribuição de servidores Apache no mundo:

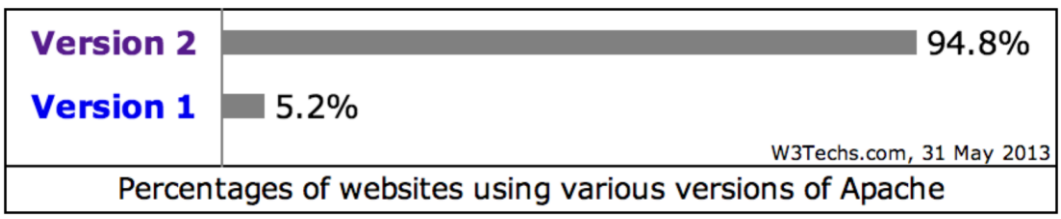

FIG 5.10 – Quantidade de servidores Apache (W3TECHS, 2013)

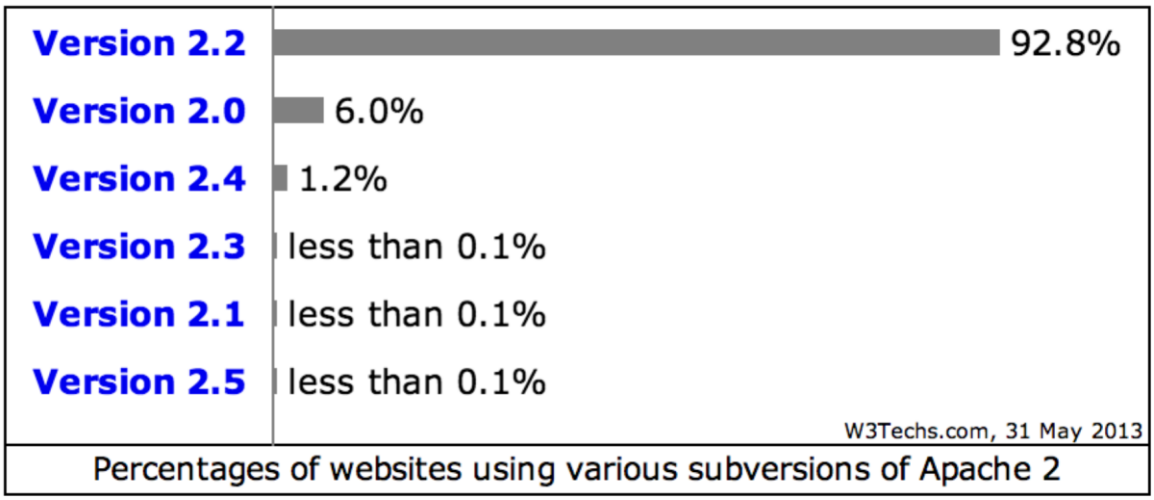

FIG 5.11 – Quantidade de servidores Apache 2.x (W3TECHS, 2013)

Os dados das figuras 5.10 informam que, no momento em que este trabalho foi realizado, junho de 2013, as versões 2.x eram as mais utilizadas. A figura 5.11 informa que, das versões 2.x, a versão 2.2 é a que ainda é a mais utilizada, seguida da versão 2.0 e 2.4.

Por meio dessas informações, foram catalogadas as vulnerabilidades que afetam as versões 2.0, 2.4 e 2.2. O Apache (2013) fornece uma lista de vulnerabilidades classificadas em 3 grupos de versões: 2.0, 2.2 e 2.4. Obteve-se um total de 143 vulnerabilidades para essas versões ao todo. A análise delas encontra-se detalhada no apêndice 10.1.

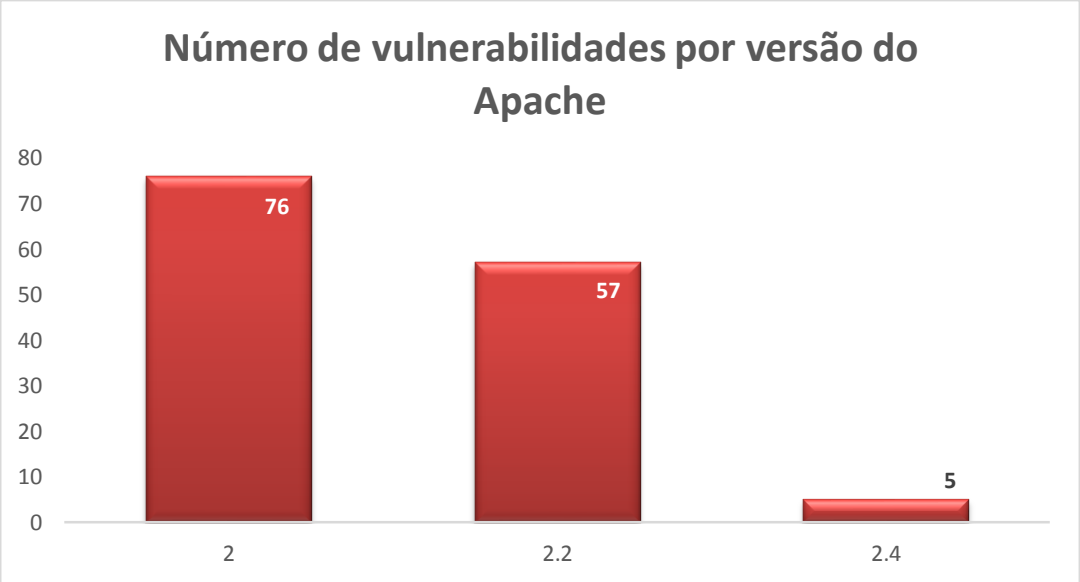

FIG 5.12- Número de vulnerabilidades por versão do Apache

Na figura acima pode ser visto o número de vulnerabilidades encontradas para as versões 2.0, 2.2 e 2.4 do Apache. A versão 2.0 apresenta o maior número de vulnerabilidades detectadas, setenta e seis, seguida da versão 2.2 que apresenta cinquenta e sete vulnerabilidades e apresentando o menor número de vulnerabilidades, cinco, tem a versão 2.4.

Na figura 5.13 é mostrado o número de vulnerabilidades de cada tipo do Apache 2.0. Das setenta e seis vulnerabilidades encontradas nesta versão na figura 5.13, trinta, correspondem a negação de serviço, DoS. Com ocorrências entre seis e nove, encontram-se as vulnerabilidades do tipo Execução de Código, *By-pass*, *Overflow*, Corrupção de Memória, Obtenção de Informação e XSS. Com o menor número de ocorrências, apenas três, figura a Obtenção de Privilégio. Não foram encontradas vulnerabilidades do tipo *Directory Traversal*.

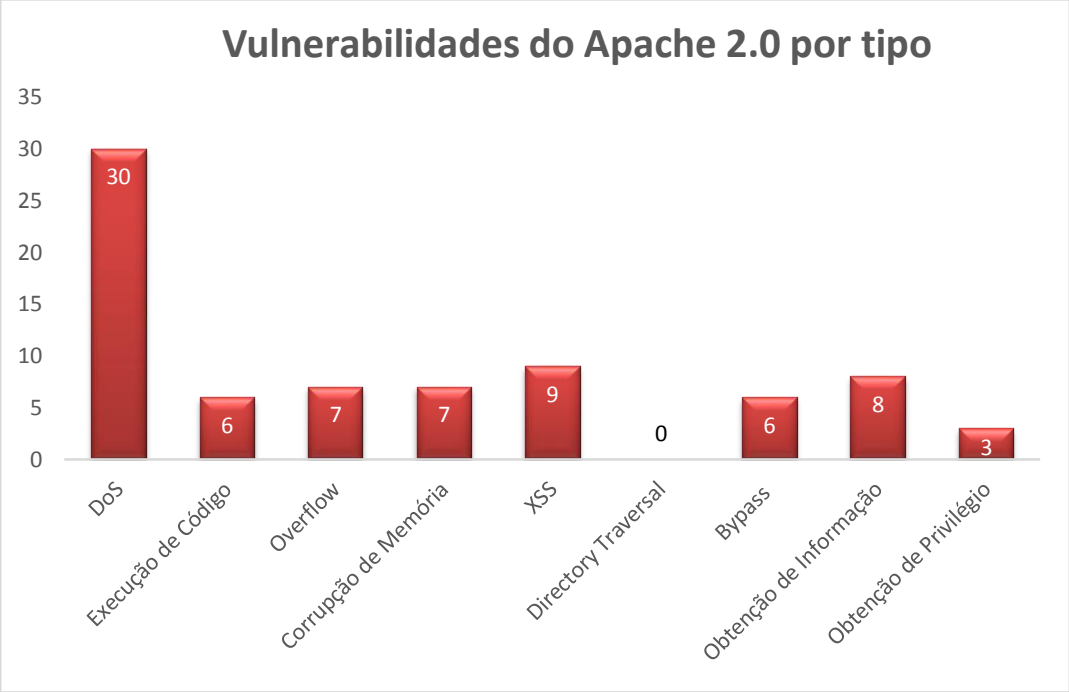

FIG 5.14 – Número de vulnerabilidades do Apache 2.0 por tipo

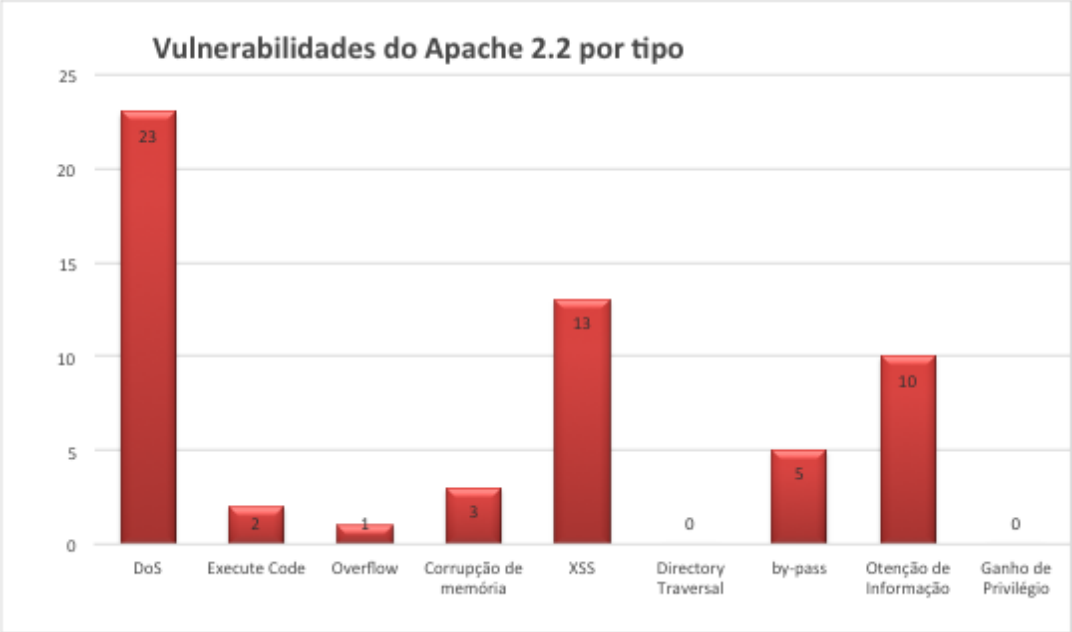

FIG 5.12 – Vulnerabilidades do Apache 2.2

A figura 5.15, mostra que, como na versão 2.0, na versão 2.2 o maior número de ocorrências de vulnerabilidades são do tipo DoS, as vulnerabilidades do tipo XSS aparecem com a segunda maior ocorrência, seguida do tipo Obtenção de Informação. Execução de Código, *Overflow*, Corrupção de Memória e *By-pass* aparecem em poucos casos, já *Directory Traversal* e Obtenção de Privilégio, não figuraram.

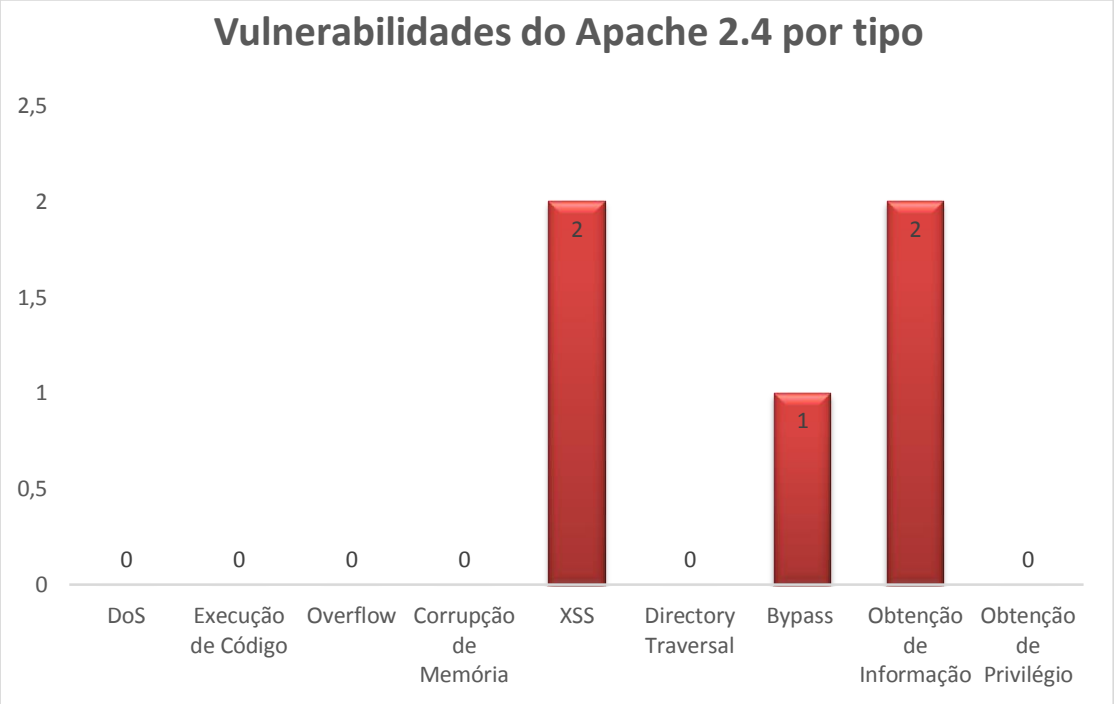

FIG 5.13 - Vulnerabilidades do Apache 2.4

Na versão 2.4 foram encontradas apenas cinco vulnerabilidades. Dessas, duas são do tipo XSS e duas são do tipo Obtenção de Informação. Foi encontrada uma vulnerabilidade do tipo *By-pass*, esses dados podem ser verificados na figura 5.16.

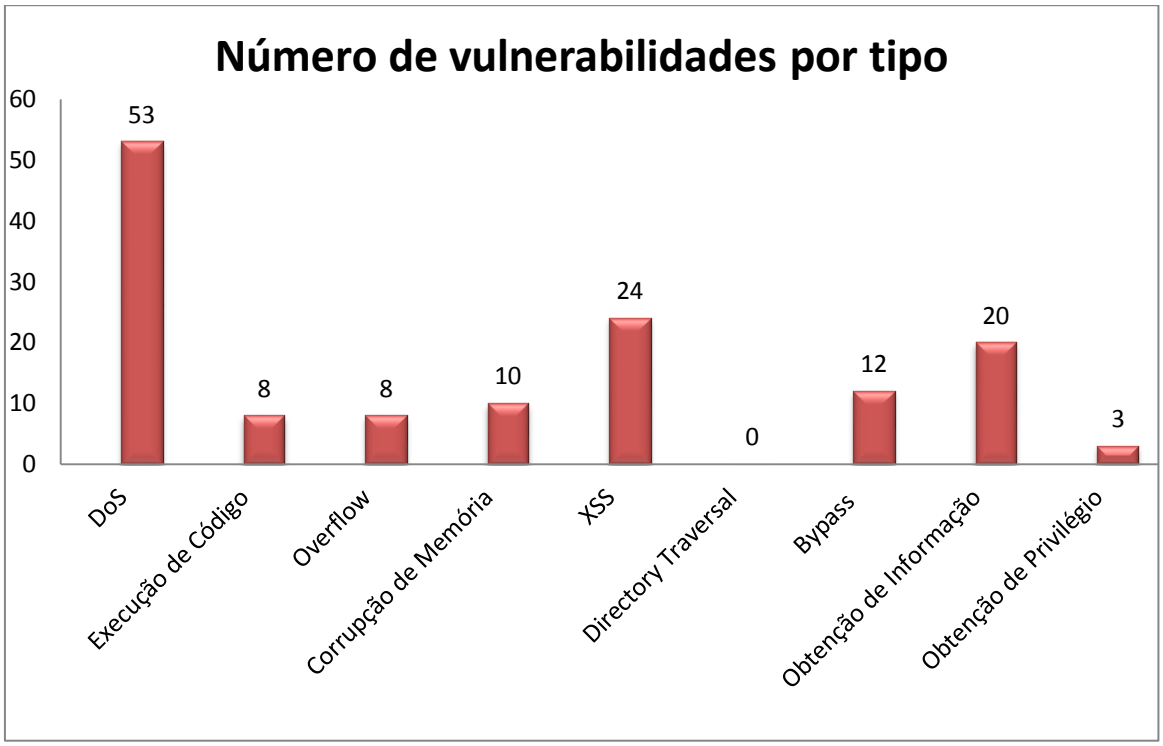

FIG 5.14 – Vulnerabilidades do Apache.

No total, entre as versões 2.0, 2.2 e 2.4 do Apache, foram encontradas cinquenta e três vulnerabilidades do tipo DoS, sendo essa a mais recorrente, seguida por vinte e quatro do tipo XSS e vinte do tipo Obtenção de Informação. Não foi encontrada nenhuma vulnerabilidade do tipo *Directory Traversal* e foram detectadas apenas três ocorrências de vulnerabilidades do tipo Obtenção de Privilégio.

5.4 IIS

A partir de uma lista de vulnerabilidades da internet feita pelo CVE® (Common Vulnerabilities and Exposures) em 2013, foi feita uma análise daquelas referentes ao IIS, desde 1999 até 2012. Foram encontradas 99 vulnerabilidades que foram classificadas de

duas maneiras distintas, primeiramente de acordo com a versão do IIS as quais elas são referentes e posteriormente de acordo com o tipo de vulnerabilidade.

Utilizando os dados disponíveis nas tabelas do apêndice elaboradas com essas classificações descritas acima e disponíveis no apêndice deste trabalho foram gerados alguns gráficos para uma melhor análise desses dados. A Figura 5.15 abaixo é o primeiro destes gráficos, referente à classificação por versão do IIS.

Figura 5.15 – Número de vulnerabilidades por versão do IIS

A partir da figura podemos ver que as versões 4.0 e 5.0 são as duas que possuem um maior número de vulnerabilidades, chegando a ter mais do que o dobro de vulnerabilidades da versão 5.1 que é a que possui o terceiro maior número.

A Figura 5.16 é o gráfico das vulnerabilidades classificadas por tipo.

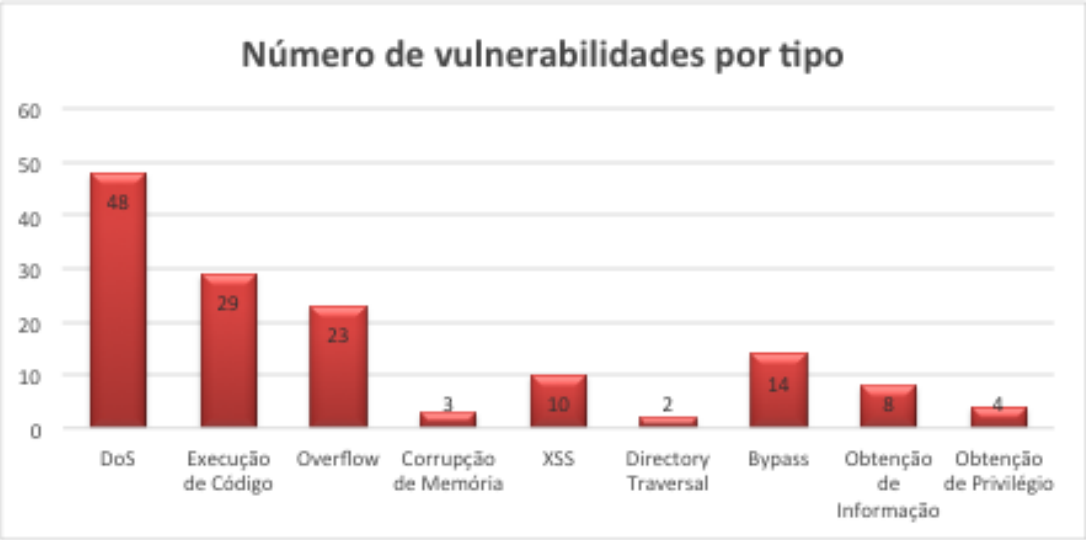

Figura 5.16 – Número de vulnerabilidades por tipo

Analisando esse gráfico acima podemos ver que o problema mais recorrente em relação as vulnerabilidades dos servidores web IIS é o de *DoS*.

Um problema que se torna aparente é o que a maioria das vulnerabilidades classificadas acima são referentes a versões do IIS, 4.0 e 5.0, que praticamente não são utilizadas atualmente, uma vez que de acordo com uma pesquisa feita pelo W3Techs em junho de 2013, de todos os sites que utilizavam o IIS, 49.4% eram referentes à versão 6.0 e 47.8% referentes à versão 7.0. A figura 5.20 mostra o gráfico com a distribuição do mercado do IIS em relação a cada versão.

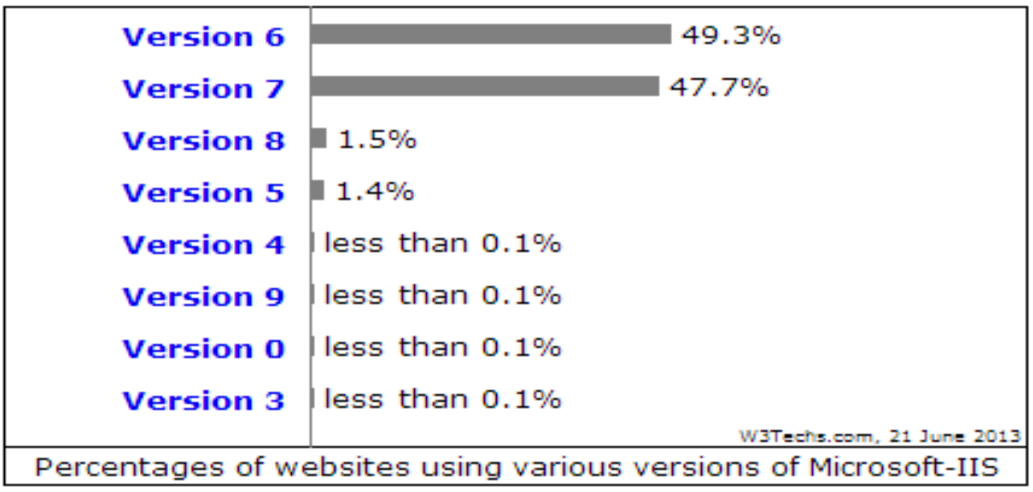

Figura 5.20 – Gráfico com distribuição do mercado do IIS.

A maioria das vulnerabilidades classificadas anteriormente não correspondem aos maiores problemas atuais, logo, se tornou necessário uma análise específica dessas duas versões. A Figura 5.1 mostra uma classificação por tipo de vulnerabilidade, considerando apenas aquelas relacionadas à versão 6.0 do IIS.

Figura 5.21 – Vulnerabilidades do IIS 6.0 por tipo

A partir dessa classificação pode-se perceber que o tipo de vulnerabilidade encontrado em maior número é o de execução de código, mas percebe-se também que há uma boa distribuição desses tipos e que *Overflow* e *by-pass* também possuem uma quantidade relevante.

A Figura 5.7 mostra o mesmo tipo de classificação utilizado na figura 5.21, entretanto dessa vez para as vulnerabilidades da versão 7.0 do servidor web IIS.

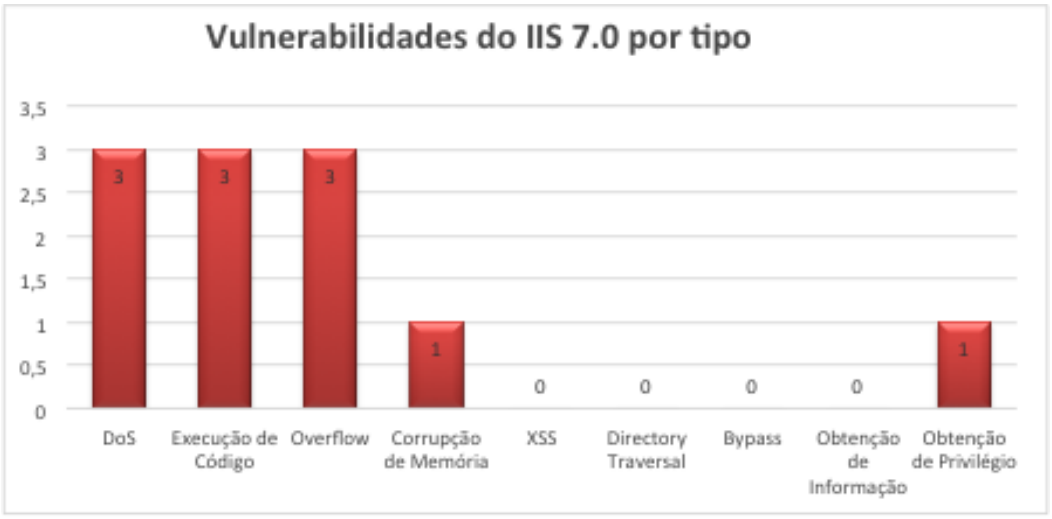

Figura 5.7 – Vulnerabilidades do IIS 7.0 por tipo

Analisando esse gráfico, pode-se notar que não há o domínio de um tipo de vulnerabilidades assim como ocorreu na primeira classificação feita dessa maneira, mas sim um domínio de três tipos que se dividem igualmente, sendo eles: *DoS*, *Overflow* e execução de código

Na figura 5.23 foi feita uma comparação entre as duas versões mais utilizadas de cada um dos softwares IIS e Apache.

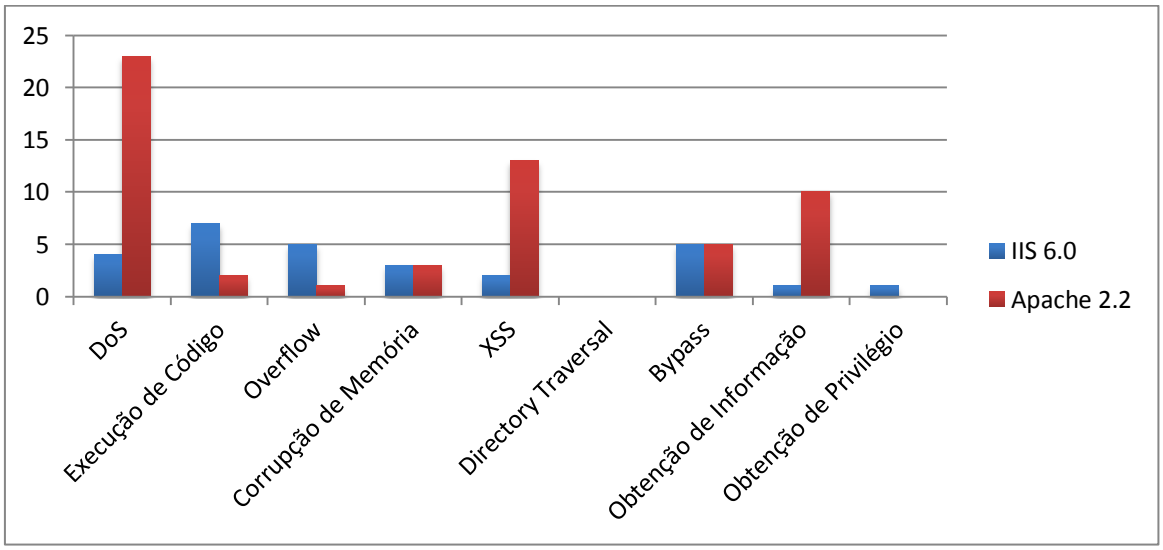

Figura 5.23 – Análise comparativa do IIS 6.0 e Apache 2.2

A partir da análise desse gráfico pode-se perceber que os softwares se diferem bastantes no que se refere às suas principais vulnerabilidades. Negação de Serviço, por

exemplo, que é um dos problemas mais frequentes quando analisado um contexto histórico completo, se destaca bastante no Apache 2.2, o que não ocorre na versão 6.0 do IIS. O mesmo podemos dizer do XSS e da obtenção de privilégio, que apesar de não serem tão frequentes no IIS 6.0 aparecem como uma das mais frequentes no Apache 2.2.

Por outro lado, Overflow e obtenção de código são tipos de vulnerabilidades de maior frequência no IIS 6.0 e são quase nulas no Apache 2.2.

6 CONCLUSÃO

Inicialmente foram estudados os servidores Web de um modo abrangente, visando suas características, funcionamento e distribuição no mercado. Tendo em vista que os servidores Web mais presentes atualmente são o Apache e IIS, foi então realizado um estudo específico de cada um, identificando suas vulnerabilidades. Com essa identificação, foram propostas nove categorias que englobam todas as vulnerabilidades observadas, sendo então possível catalogá-las.

A partir da análise dos gráficos das vulnerabilidades do Apache pode-se notar grande distribuição entre os tipos e versões. A vulnerabilidade mais recorrente é a do tipo de Negação de Serviço que ocorre em aproximadamente 40% das vulnerabilidades encontradas tanto na versão 2.0 e 2.2.

Na versão 2.4 foram encontradas apenas cinco vulnerabilidades: XSS, *By-pass* e Obtenção de Informação. Em nenhum caso, no servidor Web Apache, foi encontrada vulnerabilidade do tipo *Directory Traversal.* A qual ocorreu apenas no servidor IIS.

Em relação ao IIS, a partir das classificações feitas de suas vulnerabilidades, pode-se concluir que as versões com os maiores problemas relacionados às vulnerabilidades do tipo das explicadas neste trabalho são as versões 4.0 e 5.0, entretanto elas praticamente não são utilizadas atualmente, quando comparadas com outras versões.

Além disso, apesar do fato de que historicamente o maior problemas do IIS estar relacionado com Negação de Serviço, isso não pode ser generalizado para a situação atual, visto que as duas versões mais utilizadas nos dias de hoje, com quase 50% do mercado cada uma delas, possuem outras vulnerabilidades como as mais frequentes.

A versão 6.0 possui Execução de Código como seu maior problema, com 7 da totalidade de vulnerabilidades listadas apresentando riscos desse tipo, seguido por Overflow e By-pass, com 5 vulnerabilidades cada uma delas. Enquanto a versão 7.0 possui *DoS*, Execução de Código como seus maiores problemas, com cada uma delas apresentando 3 vulnerabilidades com riscos dessa categoria, de um total de 6 vulnerabilidades.

A partir do gráfico comparativo entre as versões mais utilizadas atualmente do IIS e do Apache, pode-se perceber que existem diversas diferenças entre os dois programas, como, por exemplo, um domínio de vulnerabilidades do tipo Negação de Serviço no Apache, fato que não ocorre no IIS, em sua versão 6.0. Por outro lado, Overflow e obtenção de código são os tipos de vulnerabilidades predominantes no IIS 6.0 e quase não existem na versão 2.2 do Apache.

Com essa divisão e análise, próximas pesquisas podem contar com uma divisão das vulnerabilidades dos servidores Web Apache, versões 2.0, 2.2 e 2.4, e IIS em nove categorias: Negação de Serviço, *Overflow, Directory Tranversal, By-pass*, Obtenção de Informação, Obtenção de Privilégio, XSS, Execução de Código e Corrupção de Memória. Essa divisão pode, por exemplo, ajudar sistemas a detectar certas vulnerabilidades e avisar aos usuários antes que elas ocorram ou, para futuramente, propor soluções inteligentes para tais vulnerabilidades.

7 REFERÊNCIA BIBLIOGRÁFICA

APACHE. **Security updates**. Disponível em: http:// httpd. apache.org/ security_report.html. [capturado em 30 de maio de 2013].

COMER, Douglas E., **Interligação de redes com TCP/I.** 5. ed. Rio de Janeiro: Elsevier,2006.

CVE. **Common Vulnerabilities and Exposures**. Disponível em: http://cve.mitre.org/. [capturado em 3 de junho de 2013].

CVE. **List of vulnerabilities**. Disponível em: http://cve.mitre.org/data/downloads/allitems.txt/ [capturado em em 28. de maio de 2013].

IIS. **Introduction to IIS architecture**. Disponível em: http://i1.iis.net/media/7188132/introduction-to-iis-architecture-101-OverviewOfHTTPRequest.png?cdn_id=2013- 03-12-001. [capturado em: 24 mar. 2013].

KABIR, Mohammed J.; **Apache Server 2 Bible**. 1. ed. Nova York: Hungry Minds, 2002.

KOZIEROK, Charles.M. **The TCP/IP guide.** 1. ed. California: No Starch, 2005.

KUROSE, James F., ROSS, Keith W. **Redes de computadores e a internet.** 5. ed. Rio de Janeiro: Pearson, 2011.

NETCRAFT. **April 2013 Web Server Survey**. Disponível em: http://news.netcraft.com/archives/2012/01/03/january-2012-Web-server-survey.html. [capturado em: 21 mar. 2013].

POSTEL, J. **Internet Message Protocol.** 1. ed. California: Information Science Institute, 1981.

RISTIC, Ivan. **Apache Security,** 1ª Edição, 2005, editora O'Reilly, California

TANENBAUM, Andrew S., **Redes de computadores**. 8. ed. Rio de Janeiro: Elsevier, 2003.

TECHNET. **Definition of a security vulnerability.** Disponível em: http://technet.microsoft.com/en-us/library/cc751383.aspx. [capturado em: 24 mar. 2013].

W3TECHS. **Usage statistics and market share of Apache for websites**. Disponível em: http://w3techs.com/technologies/details/ws-apache/all/all [capturado em 30 de maio de 2013]

W3TECHS. **Usage statistics and market share of Apache version 2 for websites**. Disponível em: http://w3techs.com/technologies/details/ws-apache/2/all [capturado em 30 de maio de 2013].

W3TECHS. **Usage statistics and market share of Microsoft-IIS for websites**. Disponível em: http://w3techs.com/technologies/details/ws-microsoftiis/all/all [capturado em 30 de maio de 2013].

.

8 APENDICE

8.1 VULNERABILIDADES DO APACHE

Tabela 8.1.1 – Vulnerabilidades por tipo

CVE	DoS	Execução de Código	Over flow	Corrupção de Memória	XSS	Directory Traversal	Bypass	Obtenção de Informação	Obtenção de Privilégio
2012-2531								X	
2010-3972	X	X	X						
2010-2731							X		
2010-2730		X	X						
2010-1899	X		X						
2010-1256		X		X					
2009-4444		X	X	X			X		
2009-3023		X	X	X					
2009-2521	X								
2009-1535							X		
2009-1122							X		
2008-1446		X	X						
2008-0075		X							
2008-0074									X
2007-2897	X	X					X	X	
2007-2815							X		
2006-6578		X							
2006-0026		X	X						

2005-4360	X	X							
2005-2678							X		
2005-2089					X		X		
2004-0205		X	X						
2004-0119	X	X							
2003-1567							X	X	
2003-1566								X	
2003-0718	X								
2003-0226	X								
2003-0225	X								
2003-0224		X	X						
2003-0223					X				
2003-1582					X				
2002-1908	X								
2002-1790							X		
2002-1700					X				
2002-1182	X								
2002-1181					X				
2002-0869									X
2002-0422								X	
2002-0419								X	
2002-0364		X	X						
2002-0224	X								
2002-0150	X	X	X						
2002-1744						X			
2002-	X	X	X						

0149									
2002-0148					X				
2002-0147	X	X	X						
2002-0079	X	X	X						
2002-0075					X				
2002-0074					X				
2002-0073	X								
2002-0072	X								
2002-0071	X	X	X						
2001-1243	X								
2001-1186	X								
2001-0545	X								
2001-0544	X								
2001-0508	X								
2001-0507									X
2001-0506			X						X
2001-0500		X	X						
2001-0337	X								
2001-0336	X								
2001-0334	X								
2001-0333		X				X			
2001-0151	X								
2001-0146	X								
2001-0096	X								
2000-1147		X	X						
2000-1104					X				

2000-0886		X							
2000-0884		X							
2000-0858	X								
2000-0770							X		
2000-0746					X				
2000-0649								X	
2000-0631	X								
2000-0408	X								
2000-0304	X								
2000-0258	X								
2000-0226	X		X						
2000-0167	X								
2000-0024							X		
1999-1591							X		
1999-1544	X		X						
1999-1537	X								
1999-1478	X								
1999-1376		X	X						
1999-1223	X								
1999-1148	X								
1999-1035	X								
1999-1011		X							
1999-0874	X		X						
1999-0867	X								
1999-0861								X	
1999-	X								

0449									
1999-0349	X	X	X						
1999-0281	X								
1999-0233		X							
1999-0012							X		

8.2 VULNERABILIDADES DO IIS

Tabela 8.2.1 – Vulnerabilidades do IIS por tipo

CVE	DoS	Execução de Código	Over flow	Corrupção de Memória	XSS	Directory Traversal	Bypass	Obtenção de Informação	Obtenção de Privilégio
2012-2531								X	
2010-3972	X	X	X						
2010-2731							X		
2010-2730		X	X						
2010-1899	X		X						
2010-1256		X		X					
2009-4444		X	X	X			X		
2009-3023		X	X	X					
2009-2521	X								
2009-1535							X		
2009-1122							X		
2008-1446		X	X						
2008-0075		X							
2008-0074									X
2007-2897	X	X					X	X	
2007-2815							X		
2006-6578		X							
2006-0026		X	X						
2005-4360	X	X							
2005-2678							X		
2005-2089					X		X		
2004-0205		X	X						
2004-0119	X	X							
2003-1567							X	X	

2003-1566								X	
2003-0718	X								
2003-0226	X								
2003-0225	X								
2003-0224		X	X						
2003-0223					X				
2003-1582					X				
2002-1908	X								
2002-1790							X		
2002-1700					X				
2002-1182	X								
2002-1181					X				
2002-0869									X
2002-0422								X	
2002-0419								X	
2002-0364		X	X						
2002-0224	X								
2002-0150	X	X	X						
2002-1744						X			
2002-0149	X	X	X						
2002-0148					X				
2002-0147	X	X	X						
2002-0079	X	X	X						
2002-0075					X				
2002-0074					X				
2002-0073	X								
2002-0072	X								
2002-0071	X	X	X						
2001-1243	X								
2001-1186	X								
2001-0545	X								
2001-0544	X								
2001-0508	X								
2001-0507									X
2001-0506			X						X
2001-0500		X	X						
2001-0337	X								
2001-0336	X								
2001-0334	X								
2001-0333		X				X			
2001-0151	X								
2001-0146	X								
2001-0096	X								
2000-1147		X	X						
2000-1104					X				
2000-0886		X							
2000-0884		X							
2000-0858	X								

2000-0770							X		
2000-0746					X				
2000-0649								X	
2000-0631	X								
2000-0408	X								
2000-0304	X								
2000-0258	X								
2000-0226	X		X						
2000-0167	X								
2000-0024							X		
1999-1591							X		
1999-1544	X		X						
1999-1537	X								
1999-1478	X								
1999-1376		X	X						
1999-1223	X								
1999-1148	X								
1999-1035	X								
1999-1011		X							
1999-0874	X		X						
1999-0867	X								
1999-0861								X	
1999-0449	X								
1999-0349	X	X	X						
1999-0281	X								
1999-0233		X							
1999-0012							X		

9 ANEXO

9.1 VULNERABILIDADES DO APACHE

Tabela 91.1 – Vulnerabilidade do Apache por versão

CVE	2.0	2.2	2.4
2002-1592	X		
2002-0392	X		
2002-0654	X		
2002-0661	X		
2002-1593	X		
2002-1156	X		
2002-0840	X		
2003-0017	X		
2003-0016	X		
2003-0132	X		
2003-0083	X		
2003-0134	X		
2003-0189	X		
2003-0245	X		
2003-0254	X		
2003-0192	X		
2003-0253	X		
2003-0789	X		
2003-0542	X		
2003-0020	X		
2004-0113	X		
2004-0174	X		
2004-0488	X		
2004-0493	X		
2004-0809	X		
2004-0751	X		
2004-0747	X		
2004-0748	X		
2004-0786	X		
2004-0811	X		
2004-0885	X		
2004-1834	X		

2004-0942	X		
2005-2088	X		
2005-2728	X		
2005-1268	X		
2005-2491	X		
2005-2970	X		
2005-2700	X		
2005-3352	X		
2005-3357	X		
2006-3747	X		
2007-1863	X		
2007-3304	X		
2006-5752	X		
2007-3847	X		
2007-5000	X		
2007-6388	X		
2008-0005	X		
2008-2364	X		
2008-2939	X		
2010-0434	X		
2009-3094	X		
2009-3095	X		
2009-1891	X		
2009-2412	X		
2010-1452	X		
2010-1623	X		
2009-3560	X		
2009-3720	X		
2010-0425	X		
2011-0419	X		
2011-3368	X		
2011-3192	X		
2012-3499		X	
2012-4558		X	
2012-2687		X	
2012-0883		X	
2012-4557		X	
2011-3607		X	
2012-0021		X	
2012-0031		X	
2011-4317		X	
2012-0053		X	
2011-3368		X	
2011-3348		X	
2011-3192		X	
2011-0419		X	
2009-3720		X	
2009-3560		X	

2010-1623		X	
2010-2068		X	
2010-1452		X	
2010-0425		X	
2010-0434		X	
2010-0408		X	
2009-3094		X	
2009-3095		X	
2009-2699		X	
2009-2412		X	
2009-1890		X	
2009-1191		X	
2009-1891		X	
2009-1195		X	
2008-0456		X	
2009-1956		X	
2009-1955		X	
2009-0023		X	
2010-2791		X	
2008-2939		X	
2007-6420		X	
2008-2364		X	
2008-0005		X	
2007-6422		X	
2007-6421		X	
2007-6388		X	
2007-5000		X	
2007-3847		X	
2006-5752		X	
2007-3304		X	
2007-1862		X	
2007-1863		X	
2006-3747		X	
2005-3357		X	
2005-3352		X	
2012-3499			X
2012-4558			X
2012-3502			X
2012-2687			X
2012-0883			X

9.2 VULNERABILIDADES DO IIS

Tabela 9.2.2 – Vulnerabilidades do IIS por versão

CVE	1	2	3	4	5	5.1	6	7	7.5
2012-2531									X
2010-3972								X	X
2010-2731					X				
2010-2730									X
2010-1899						X	X	X	X
2010-1256							X	X	X
2009-4444					X	X	X		
2009-3023					X		X		
2009-2521					X		X	X	
2009-1535					X	X	X		
2009-1122					X				
2008-1446					X	X	X	X	
2008-0075					X	X	X		
2008-0074					X	X	X	X	
2007-2897							X		
2007-2815					X				
2006-6578						X			
2006-0026					X	X	X		
2005-4360						X			
2005-2678						X	X		
2005-2089					X		X		
2004-0205				X					
2004-0119				X					
2003-1567					X				
2003-1566					X				
2003-0718					X	X	X		
2003-0226					X	X			
2003-0225				X	X				
2003-0224					X				
2003-0223				X	X	X			
2003-1582							X		
2002-1908					X	X			
2002-1790				X	X				
2002-1700					X				
2002-1182					X	X			
2002-1181				X	X				
2002-0869				X	X	X			

2002-0422					X	X			
2002-0419				X	X	X			
2002-0364				X	X				
2002-0224					X				
2002-0150				X	X	X			
2002-1744					X				
2002-0149				X	X	X			
2002-0148				X	X	X			
2002-0147						X			
2002-0079				X	X				
2002-0075				X	X	X			
2002-0074				X	X	X			
2002-0073				X	X	X			
2002-0072				X	X	X			
2002-0071				X	X				
2001-1243				X	X				
2001-1186					X				
2001-0545				X					
2001-0544					X				
2001-0508					X				
2001-0507					X				
2001-0506				X	X				
2001-0500	X	X	X	X	X	X			
2001-0337	X	X	X	X	X				
2001-0336	X	X	X	X	X				
2001-0334	X	X	X	X	X				
2001-0333	X	X	X	X	X				
2001-0151					X				
2001-0146					X				
2001-0096				X	X				
2000-1147				X					
2000-1104				X	X				
2000-0886					X				
2000-0884				X	X				
2000-0858				X					
2000-0770				X	X				
2000-0746				X	X				
2000-0649				X					
2000-0631			X	X	X				
2000-0408				X	X				
2000-0304				X	X				
2000-0258				X	X				
2000-0226				X					
2000-0167				X					
2000-0024				X					
1999-1591				X					
1999-1544			X	X					
1999-1537			X	X					

1999-1478			X	X					
1999-1376				X					
1999-1223			X						
1999-1148	X	X	X	X					
1999-1035			X	X					
1999-1011			X	X					
1999-0874				X					
1999-0867				X					
1999-0861				X					
1999-0449				X					
1999-0349			X	X					
1999-0281		X	X						
1999-0233	X								
1999-0012				X					