

Unix System Administration

Chris Schenk

Lecture 16 – Thursday Mar 06

CSCI 4113, Spring 2008

Message Authentication Codes

- MACs
 - Not to be confused with network Layer 1 addresses!
- MACs use hash functions to tag messages
 - Used by our RNDC utility for BIND
 - All messages are authenticated with a 'tag'
- Not the same as a digital signature!
 - Digital sigs use assymmetric keys
 - MACs use a symmetric key with the hash

SSL Cert Creation

- Using an RSA key, we create a certificate signing request (CSR)
 - Contains email, common name (CN), and others
 - Plain-text info is hashed (usually with MD5 or SHA1)
 - Hash is encrypted with RSA private key
 - Self-signed so the CA knows we own private key
 - Contains no expiration date
- CA verifies who we are
 - Removes (?) self-signature, adds in their own
 - Adds in expiration date and 'issuer' information

SSL Cert Verification

- Browser is presented with SSL cert
 - Checks expiration date
 - Checks if Common Name (CN) matches URL
 - Browser finds CA root cert in browser that matches 'issuer' field
- Decode CA signature using CA's public key
 - Contained in the CA root cert
 - Hash plain-text data and compare hashes
- If all three criteria are met, then continue
 - Otherwise create a pop-up window

Web – Apache Web Server

- Apache runs as `httpd` or `apache2` daemon
 - depending on the distro
 - 'apache2' under Ubuntu
- Highly configurable
 - Allow/disallow dynamic content with PHP, Perl, Python, etc
 - Fine-grained permissions and options per directory
 - User-defined web pages from `public_html` dirs
 - SSL support enabled with a module

Apache – Ubuntu File Locations

- Any of the following locations is configurable
 - I generally like to use the defaults, however
- `/etc/apache2/apache2.conf` – Main config
- `/var/www` – Default location for webserver files
- `/usr/sbin/apache2` – Daemon
- `/var/log/apache2` – Log file location
 - Many different logs exist in this directory
- `/etc/init.d/apache2` – Startup script

Ubuntu File Locations

- Sites-available vs sites-enabled
 - A way to easily deploy or disable websites configured on the webserver
- `/etc/apache2/sites-available`
 - Sites that can be enabled or disabled
 - Each site has its own configuration file that defines the site
- `/etc/apache2/sites-enabled`
 - symbolic links to config files in sites-available
- All enabled sites are included in `apache2.conf`

Apache – Server Directives

- Tons of different options, we'll cover important ones
 - `ServerRoot` – Location of configs/modules
 - `DocumentRoot` – Location of webserver files
 - `Listen <port>` – Bind port for apache (80)
 - `LoadModule` – Loads extra useful modules
 - `DirectoryIndex` – Default files to load in a directory (index.html, index.php, etc)
 - `User/Group` – What users to run as non-privileged
 - `ServerAdmin` – Contact email address (usually webmaster)

Apache – Directory Permissions

- Set with the `Directory` directive
- ```
<Directory />
 Options None #Turn all features off
 AllowOverride none #disable .htaccess
 Order allow,deny
 Deny from all #deny access to files
</Directory>
```
- The above disables all options and denies all access to /
  - This is root on the FILESYSTEM, not the webserver docroot!
  - Useful to set restrictive permissions on a site

# Apache – Directory Permissions (cont)

- Now we can set the permissions on the document root
- ```
<Directory /var/www>  
    #set default options  
    Options Indexes SymLinksIfOwnerMatch  
    #allow .htaccess to override default options  
    AllowOverride All  
    Order allow,deny  
    Allow from all  
</Directory>
```
- This allows the site to function for files under `/var/www`
 - Because of the `Allow from all` directive

Virtual Hosts

- Once NameVirtualHost is enabled one must configure VirtualHost's to be matched

- ```
NameVirtualHost 128.138.202.101:80
<VirtualHost 128.138.202.101:80>
 ServerName www.zipzoomfly.com:80
 ServerAdmin webmaster@zipzoomfly.com
 DocumentRoot /var/www/zipzoomfly
 ...
</VirtualHost>
<VirtualHost 128.138.202.101:80>
 ServerName www.bingo.com:80
 ServerAdmin webmaster@bingo.com
 DocumentRoot /var/www/bingo
 ...
</VirtualHost>
```

- VirtualHost IP addresses must match the NameVirtualHost IP address

# Apache NameVirtualHost'ing

- Can host multiple domain names
  - Very much like how BIND can host multiple zones
  - Known as a `NameVirtualHost` in Apache
  - Apache matches the virtual host to present to user based on host presented in URL request from user
- Hosting companies use a single web server
  - And have multiple domains hosted on it
  - godaddy.com as one example (and tons more)
- Named virtual hosts **BREAK** ssl!
  - SSL is tied to the hostname used in the URL

# How NameVirtualHost Breaks SSL

- SSL configuration on a webserver is tied to a specific IP address and port (usually 443)
  - This (usually) means one instance of SSL per server
  - We can only configure one cert per port!
- SSL checks the following for a valid SSL site:
  - Signature on cert verified by CA root cert
  - Cert has not expired
  - Common Name (CN) within cert matches URL name
- When the CN is different, we get a popup