

Wrapping CGI Scripts

Managing CGI Development Security on Apache

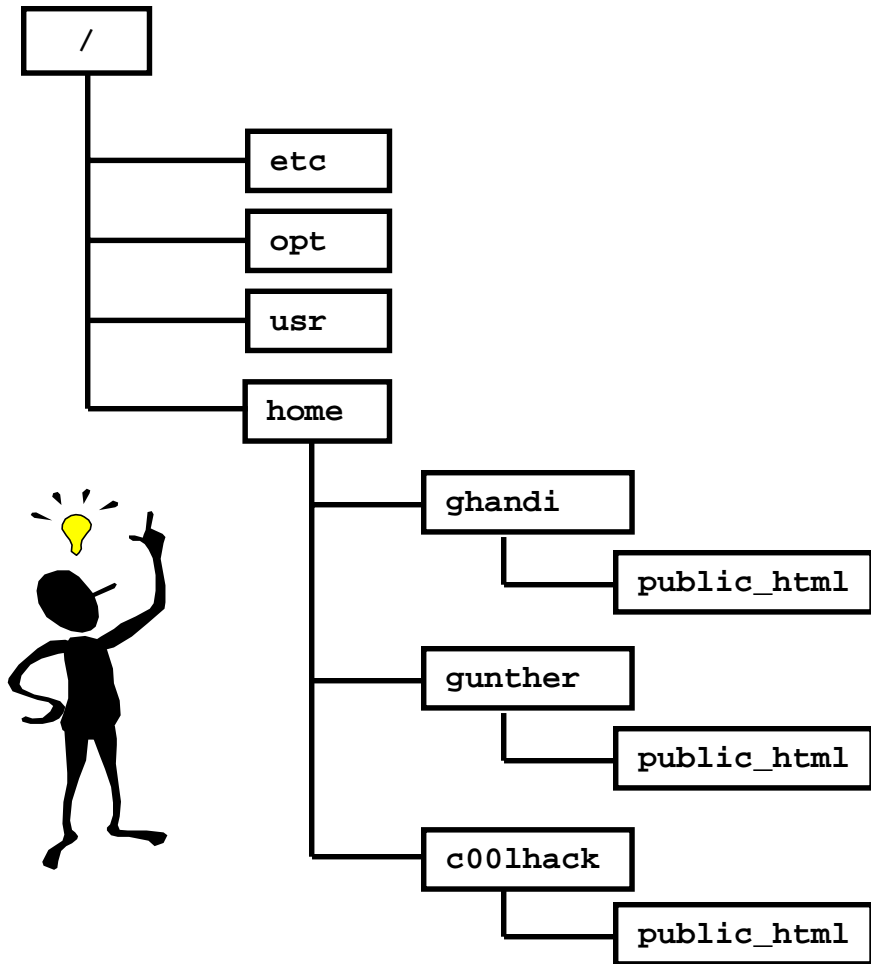
Gunther Birznieks
Gunther@eXtropia.com
<http://www.eXtropia.com/presentations/>

Wrapping CGI Scripts

- The Problem
 - Internal Web Developers
 - Should not be able to view another group's confidential documents
 - Should not be able to affect another group's work
 - Examples of shared environments include ISPs, large corporations, universities



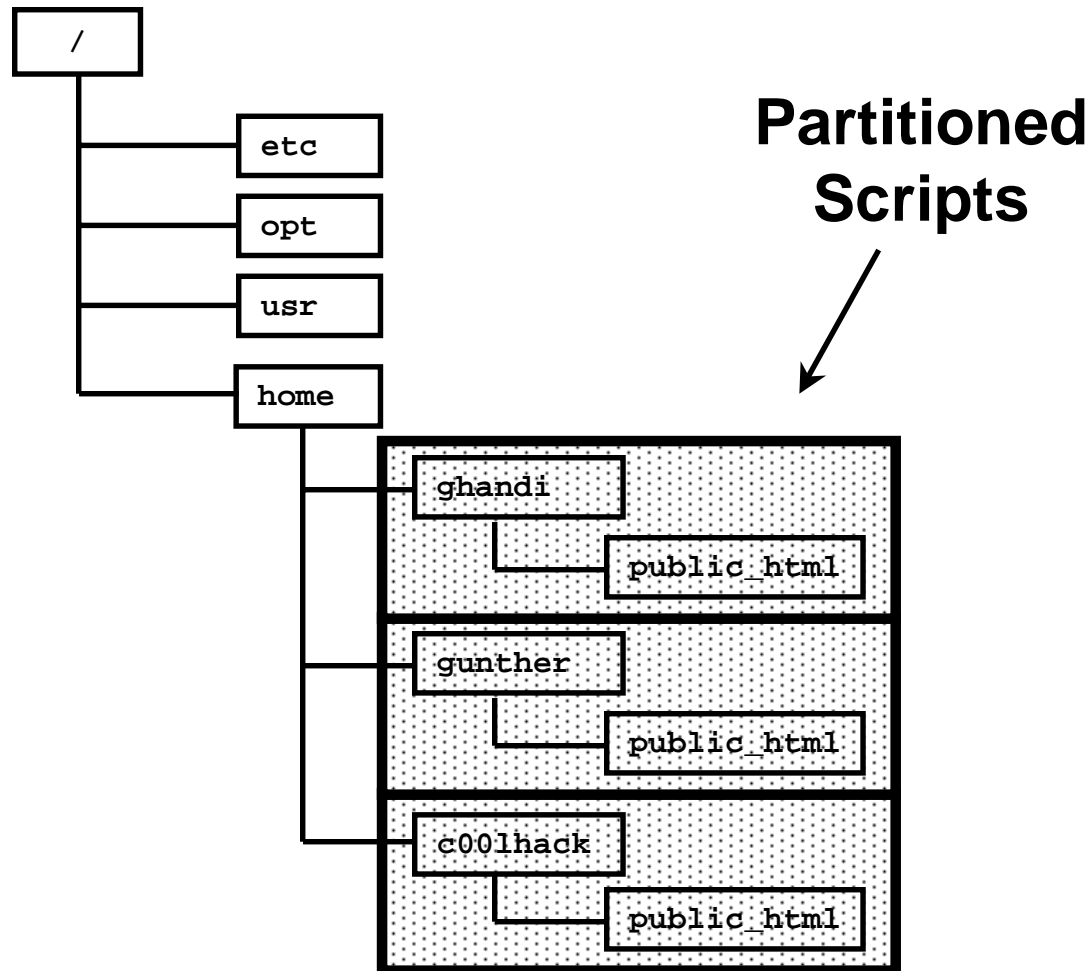
Wrapping CGI Scripts



Wrapping CGI Scripts

- Solution - Use Development “Firewalls”
 - Full Wrappers
 - Apache “built-in” suEXEC
 - cgiwrap
 - sbox
 - Partitioning with Several Web Servers

Wrapping CGI Scripts



Wrapping CGI Scripts

- Wrappers are not a panacea for external security
 - Still need to secure server from outside attacks
 - Is the language secure?
 - Perl's -T taintmode flag, Servlet security model
 - <http://www.extropia.com/tutorials/taintmode.html>
 - Is the CGI securely programmed?
 - Check input for special chars, buffer overflows
 - Lincoln Stein's WWW Security FAQ
 - <http://www.w3.org/Security/Faq/www-security-faq.html>

Wrapping CGI Scripts

- However, wrappers can help with external security
 - At least Developer A's security hole will not compromise Developer B's scripts
- Let's look at some development scenarios...

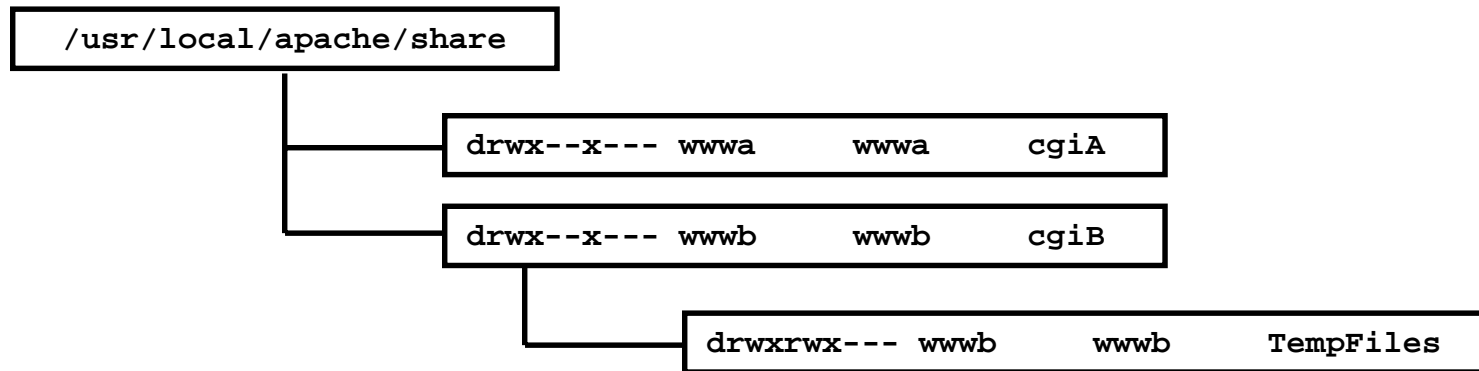
Developer Scenario

- Assumptions
 - Already use UNIX Security to partition developers
 - Use permissions layout from Lincoln Stein's Web Security book
 - Slightly more secure than Apache and WWW Security FAQ recommendations
 - Web Server runs as unprivileged user (eg www)
 - Dealing with two development groups wwwA, wwwB

Developer Scenario

- /cgiA
 - owned by wwwA, rwx permissions
 - group by wwwA, x permissions
 - world gets no permissions
- /cgiB
 - owned by wwwB, rwx permissions
 - group by wwwB, x permissions
 - world gets no permissions

Developer Scenario



Note: Web Server runs as user www, belongs to wwwa,wwwb groups.

Developer Scenario

- In order for scripts to execute, www user must belong in the wwwA and wwwB groups.
 - Afterall, the web server needs to be able to execute the scripts
- wwwA cannot access wwwB files
 - and vice versa
- But...
 - because www user belongs to both groups, wwwA user can write a script to peek into wwwB files

Developer Scenario

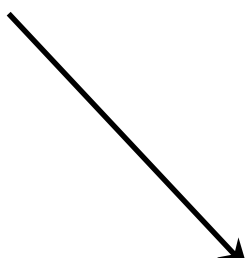
- Worse...
 - If some CGI directories are writable, these may become corrupted by another developer
 - For example
 - Shopping Cart files
 - BBS or Chat messages
 - Calendar files

Developer Scenario 2

- Let's consider the same scenario with some additions
 - /docsA, /docsB directories added
 - Assume that these are Web Server password protected directories.
- CGI scripts from wwwA or wwwB can read these files and totally bypass the security

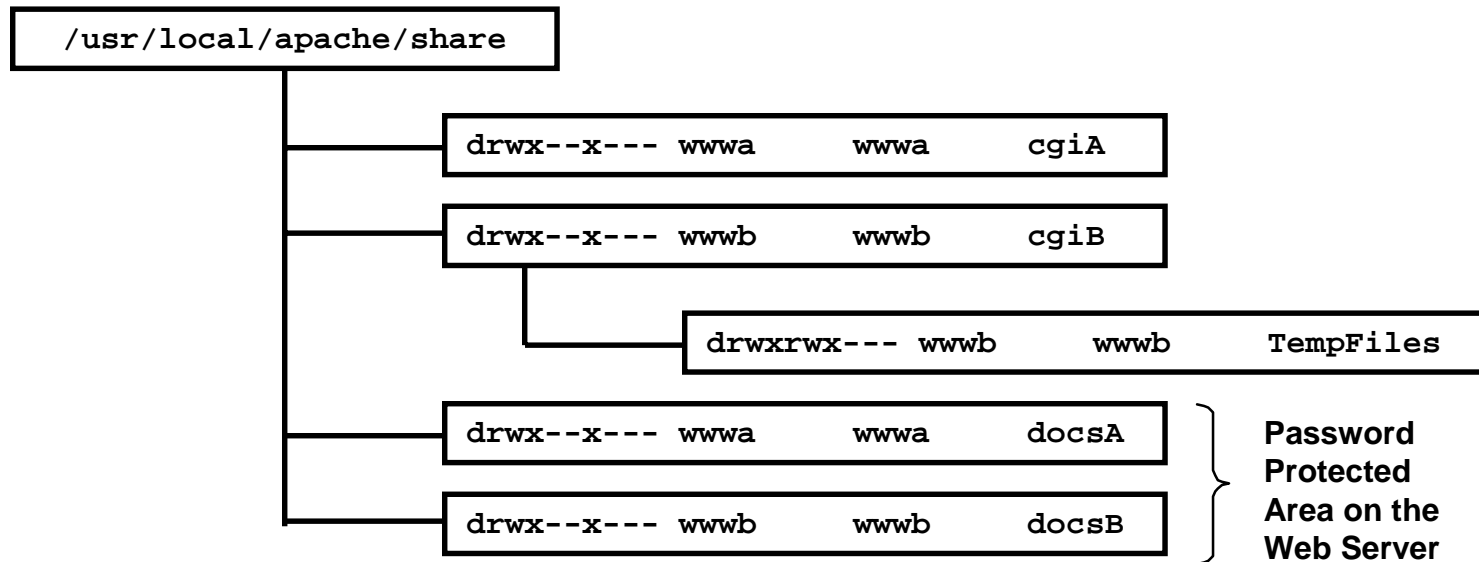
Developer Scenario 2

```
#!/usr/local/bin/perl -T  
  
use CGI;  
$query = new CGI;  
  
$base = "/Apache/docsA"  
print $query->header();  
  
open (IN, "<$base/$ENV{PATH_INFO}");  
while (<IN>) { print $_; }  
close (IN);
```



**Taint mode
won't help
us here...**

Developer Scenario 2



Partitioning Strategies

- Wrappers
- Multiple Web Servers

Partitioning with CGI Wrappers

- Advantages
 - Uses a single web server
 - Lower administrative overhead
- Disadvantages
 - Each CGI script launches two processes
 - The wrapper + the script
 - Leaves out protection for server side services such as mod_perl, jserv java servlets
 - suid programs running as root are dangerous!

Partitioning with Several Web Servers

- Advantages
 - mod_perl, jserv, etc. inherit the server permissions
 - single process
- Disadvantages
 - Can be difficult to maintain with many web servers
 - multiple log files
 - multiple configuration files
 - Can be difficult to tune
 - How many engines devoted to each user?

Available Wrappers

- suEXEC
 - Integrated with Apache
- cgiwrap v3.6.2
 - Nathan Neulinger
 - <http://www.cgi.umr.edu/~cgiwrap/>
- sbox v.98
 - Lincoln Stein
 - <http://stein.cshl.org/~lstein/sbox/>

suEXEC

- Advantages
 - Integrated with Apache
 - Supports <VIRTUAL> host sections
 - Supports user directories
- Disadvantages
 - Limited options - this can be a good thing though!
 - We'll look at the additional options of `sbox`, `cgiwrap` later...
 - No special debugging features except `cgi.log`
 - No resource limit checking

suEXEC Checks

- Let's walk through the security checks
 - Called with 3 arguments
 - target user, target group, CGI script
 - Checks to see if running as apache user
 - Also has to be a valid user
 - Program must not have root / or .. References

suEXEC Checks

- Target user, Target group is valid, matches program user, group
- Target user, group above minimum uid,gid, not superuser
- Directory, Program cannot be writable by anyone else
- Target program not setuid, setgid
- Also cleans the path (/usr/bin:/usr/local/bin)
- Environment cleanses of non-CGI variables

CGIWRAP

- Advantages
 - It's been around for a long time so its well tested
 - EXCELLENT Debugging features
 - use cgiwrapd instead of cgiwrap
 - Integrates with AFS Security (Andrew File System)
 - Access/Deny files
 - user@xxx.xxx.xxx.xxx/yyy.yyy.yyy.yyy
 - xxx=domain, yyy=mask
- Disadvantages
 - Very user-directory-centric

CGIWRAP Checks

- Called directly
 - Uses PATH_INFO, SCRIPT_NAME environment variables
 - First part of PATH_INFO is the user name
- Checks to see if running as apache user
- Program must not have root / or .. References

CGIWRAP Checks

- Target user, Target group is valid, matches program user, group
 - Target user info retrieved from PATH_INFO
- Target user, group above minimum uid,gid, not superuser
- [+] Check For Symlink script

CGIWRAP Checks

- Directory, Program cannot be writable by anyone else
- Target program not setuid, setgid
- Also cleans the path (/usr/bin:/usr/local/bin)
 - OPTIONAL, *NOT* set by default
- [-] Does not cleanse Environment of non-CGI variables
- Sets of defined resource limits such as CPU time, memory usage, etc. to prevent denial of service attacks

CGIWRAP Checks

- [+] Log script execution
 - Syslog
 - Regular log
- [+] Optional check for script not in subdirs

CGIWRAP Debugging

Environment Variables:

```
QUERY_STRING: ''
SCRIPT_NAME: '/cgi-bin/cgiwrapd'
PATH_INFO: '/~gunther/stuff/test.cgi'
PATH_TRANSLATED: '/home/gunther/public_html/stuff/test.cgi'
REMOTE_USER: '<NULL>'
REMOTE_HOST: '<NULL>'
REMOTE_ADDR: '10.1.1.1'
```

Trying to extract user from PATH_INFO.

Retrieved User Name: 'gunther'

User Data Retrieved:

```
UserID: 'gunther'
UID: '501'
```

CGIWRAP Debugging

GID: '501' Home Dir: '/home/gunther'

Script Base Directory: '/home/gunther/public_html/cgi-bin'

Trying to extract script from PATH_INFO

Script Relative Path: 'stuff/test.cgi'

Script Absolute Path: '/home/gunther/...'

Fixing Environment Variables.

Environment Variables:

QUERY_STRING: ''

SCRIPT_NAME: '/cgi-bin/cgiwrapd/gunther/stuff/test.cgi'

PATH_INFO: ''

PATH_TRANSLATED: '/usr/local/apache/share/htdocs'

REMOTE_USER: '<NULL>'

REMOTE_HOST: '<NULL>'

REMOTE_ADDR: '10.1.1.1'

Wrapping CGI Scripts - Gunther Birznieks

CGIWRAP Debugging

UIDs/GIDs Changed To:

RUID: '501'

EUID: '501'

RGID: '501'

EGID: '501'

Changing current directory to '/home/gunther/...'

Output of script follows:

=====

Content-type: text/html

<HTML><H1>/home/gunther/public_html/cgi-bin/stuff</H1></HTML>

SBOX

- Advantages
 - Performs chroot at the USER level
 - Can be set to do setgid instead of or in addition to setuid
 - Can be set to do setuid/setgid based on directory instead of program
- Disadvantages
 - Experimental -- Use at your own risk!
 - CGI current working directory problem.

SBOX Checks

- Called directly
 - Uses PATH_INFO, PATH_TRANSLATED environment to get script location
- Checks to see if running as apache user
- [+] Checks to see if running as web server group
- Program must not have root / or .. References

SBOX Checks

- Only checks program user, group above min uid, gid not superuser
 - Concept of target user, group does not exist in sbox
- Directory, Program cannot be writable by anyone else
- [+] Performs chroot on user directory
- Sets of defined resource limits such as CPU time, memory usage, etc.

SBOX Checks

- Target program not setuid, setgid
- Also cleans the path (/usr/bin:/usr/local/bin)
- Environment cleanses of non-CGI variables

Integrating a Separate Wrapper in Apache

- sbox/cgiwrap disadvantage
 - The URLs are Ugly
 - SBOX
 - <http://www/cgi-bin/sbox/stuff/test.cgi>
 - CGIWRAP
 - <http://www/cgi-bin/cgiwrap/gunther/stuff/test.cgi>

Integrating a Separate Wrapper in Apache

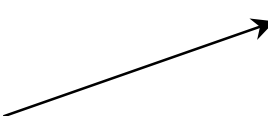
- Solution
 - use `mod_rewrite` to make the wrapper transparent
 - Sample directive below from Yugi Shinozaki in [CGIWrap Tips and Tricks](#)

```
RewriteEngine on
RewriteRule ^/~([^/]+)/cgi-bin/(.*) /cgi-bin/cgiwrap/$1/$2 [PT]
RewriteRule ^/~([^/]+)/cgi-bin-d/(.*) /cgi-bin/cgiwrapd/$1/$2 [PT]
RewriteRule ^/~([^/]+)/nph-bin/(.*) /cgi-bin/nph-cgiwrap/$1/$2 [PT]
RewriteRule ^/~([^/]+)/nph-bin-d/(.*) /cgi-bin/nph-cgiwrapd/$1/$2 [PT]
```

Integrating a Separate Wrapper in Apache

- Add authentication support
 - Either make an auth-cgi-bin separately and put cgiwrap in there or...
- Specify a different mod_rewrite rule...

```
RewriteEngine on
RewriteCond %{LA-U:REMOTE_USER} .+
RewriteRule ^/~([^/]+)/cgi-bin/(.*) /cgi-bin/cgiwrap/$1/$2 [PT,NS]
```



Forces Authentication
Check To Occur Before
Rule Rewrite

Integrating a Separate Wrapper in Apache

- Example for sbox...

```
RewriteEngine on
RewriteCond %{LA-U:REMOTE_USER} .+
RewriteRule ^/cgi-bin/(.*) /cgi-bin/sbox/$1 [PT,NS]
```

Partitioning with Several Web Servers

- Problems
 - User sees many web servers
 - different ports/hostnames are ugly
 - Log files are hard to synchronize
 - Configuration files are a pain to maintain
- Use Apache tricks to make administration easier

Partitioning with Several Web Servers

- To solve ports/hostname difficulty, use a front-end web server...
 - Compile in the following modules
 - mod_proxy
 - mod_rewrite

```
RewriteEngine on
```

```
RewriteRule ^/(gunther/cgi-bin/.* ) http://gunther.domain.com/$1 [P,L]
```

```
ProxyPassReverse / http://gunther.domain.com/$1
```



Required for redirect messages

Partitioning with Several Web Servers

- To synchronize log files, try logging to a database or syslog
- Lincoln Stein's Perl98 Cool Tips with Apache Talk
 - <http://stein.cshl.org/~lstein/talks/perl/perl98.html>
 - Code on the following page...

Partitioning with Several Web Servers

Add to httpd.conf:

```
CustomLog "| /usr/local/apache/bin/logger gunther" common
```

logger program:

```
#!/usr/local/bin/perl
# script: logger
use Sys::Syslog;
$SERVER_NAME = shift || 'www';
$FACILITY = 'local0';
$PRIORITY = 'info';
Sys::Syslog::setlogsock('unix');
openlog ($SERVER_NAME, 'ndelay', $FACILITY);
while (<>) {
    chomp;
    syslog($PRIORITY, $_);
}
closelog;
```

Wrapping CGI Scripts - Gunther Birznieks

Partitioning with Several Web Servers

- Managing multiple configurations
 - Use `<IfDefine>` directive

Start particular httpd:

```
httpd -DGunther
```

Add specifics to httpd.conf:

```
<IfDefine Gunther>  
    Port 8001  
</IfDefine>
```

Summary

- suEXEC is probably the easiest wrapper but it has the least features and won't integrate with mod_perl
- CGIWRAP/Sbox provide some interesting additional features
- Wrappers are useful for ISPs where you have to maintain many different development environments
- Using multiple web servers running as different UIDs may be better for a corporation with only a limited set of separate development groups

More Information...

- Updated Talk
 - <http://http://www.extropia.com/presentations>
- Open Source Software and other links
 - <http://www.extropia.com/>
- Acknowledgements
 - Mark McDonald, Scott Clasen, Bill Lee, Anthony Masiello, Peter Chines, Erik Ferlanti