Project Lockdown
OHS Web Server Edition (712)

Tuesday, April 20 from 4:30 - 5:30

Brian J. Mulreany

Kevin Sheehan

Presenters

- Brian J. Mulreany
  - 20+ years of experience with Oracle Products
  - 10+ years of experience with Web and Java technology
  - Technical director with AT&T and Oracle Consulting focusing on software architecture
  - Senior Architect with Unisys supporting DHS
- Kevin Sheehan
  - CISSP with 29 years of IT experience
  - 16 years experience with Oracle technology
  - 7 years in Homeland Security sector
  - Technical Director at Oracle, Unisys & Agilex Technologies

IOUG Membership Benefits

- **Information**
  - **Library of Oracle Knowledge**
  - ***SELECT Journal***
  - **5 Minute Briefing**
- **Education**
  - **Collaborate Conferences**
- **Networking**
  - **Member Directory**
  - **Special Interest Groups**
  - **Discussion Forums**
- **Advocacy**

Agenda

- 1 – Setting up the Web Tier
- 2 – Advanced Hardening
- 3 – Monitoring the Web Tier
- 4 – Planning for the Future

1 - Setting up the Web Tier

- Configuring for each use case
- Stop leaking configuration information
- Don’t pick *that* OHS
- Pick the right OHS by use case
- How many user IDs to run OHS?
- Using IfDefine for Control

Don't pick *that* OHS

- There are 10+ versions of OHS

"It is externally labeled as "10.1.3.3", but the component version is actually "10.1.3.1", and is a special build, different than the Oracle Application Server counterpart."

- All OHS versions are not created equal

"Something to think about…The Oracle HTTP Server delivered with the Oracle Database 10.2 Companion CD is provided to initially get HTMLDB installed and running. However, its an older version with limited functionality and support. Both the Oracle HTTP Server and HTMLDB from this CD would need to be upgraded at this time. The Companion CD also installs a mix of 10.2 and 10.1 products which is more difficult to maintain."

Pick the right OHS by use case

OHS	Apache	When to Choose
9iAS	1.3	Required for EBS 11i deployments, no other option
10gAS	1.3	Required for EBS 12 deployment, no other option
10gAS	1.3	The 10.1.2 version is primarily used for custom Java deployments. The 10.1.3 is for SOA Suite and can also be used for OBIEE deployments.
10gAS	2.0	Best for reverse proxy use if you want to take advantage of mod_security. Works well for UCM integration.
11gDB	2.0	Only use this version for Apex if you are not ready for 11gAS or you are limited by license
11gAS	2.2	Best for general purpose use, but does not support mod_security or mod_oc4j.

How many UserIDs to run OHS?

"Two-Man Rule" or "Four-Eyes Principle"

A security control technique that requires more than one person or more than one user ID to compromise an entire system.

It takes ***three*** User IDs to run OHS.

1. One user ID to own the OHS software
2. One user ID to run the OHS web software
3. One user ID to own the web content

Using IfDefine for Control

- Controls SSO module, Restrict mode in EBS
- Use it to control proxy or other modules

```
<IfDefine RUNPROXY>
    LoadModule proxy_module modules/mod_proxy.so
</IfDefine>
```

- Set the opmn variables you need

```
<process-type id="OHS" module-id="OHS2">
  <module-data>
    <category id="start-parameters">
      <data id="start-mode" value="ssl-enabled"/>
      <data id="command-line" value="-D RUNPROXY"/>
    </category>
```

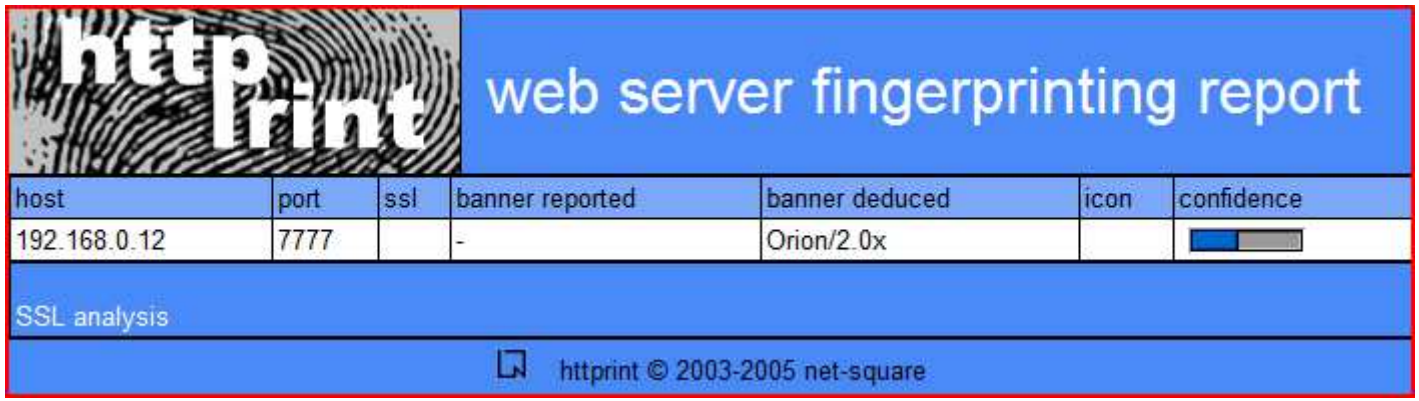
2 – Advanced Hardening

- Fingerprinting OHS
- CIS Apache Benchmark
- Configuring OHS as a reverse proxy
- Mod_status vulnerability
- Mod_security vs. mod_rewrite
- Grid Control Setup

Fingerprinting OHS Base Install

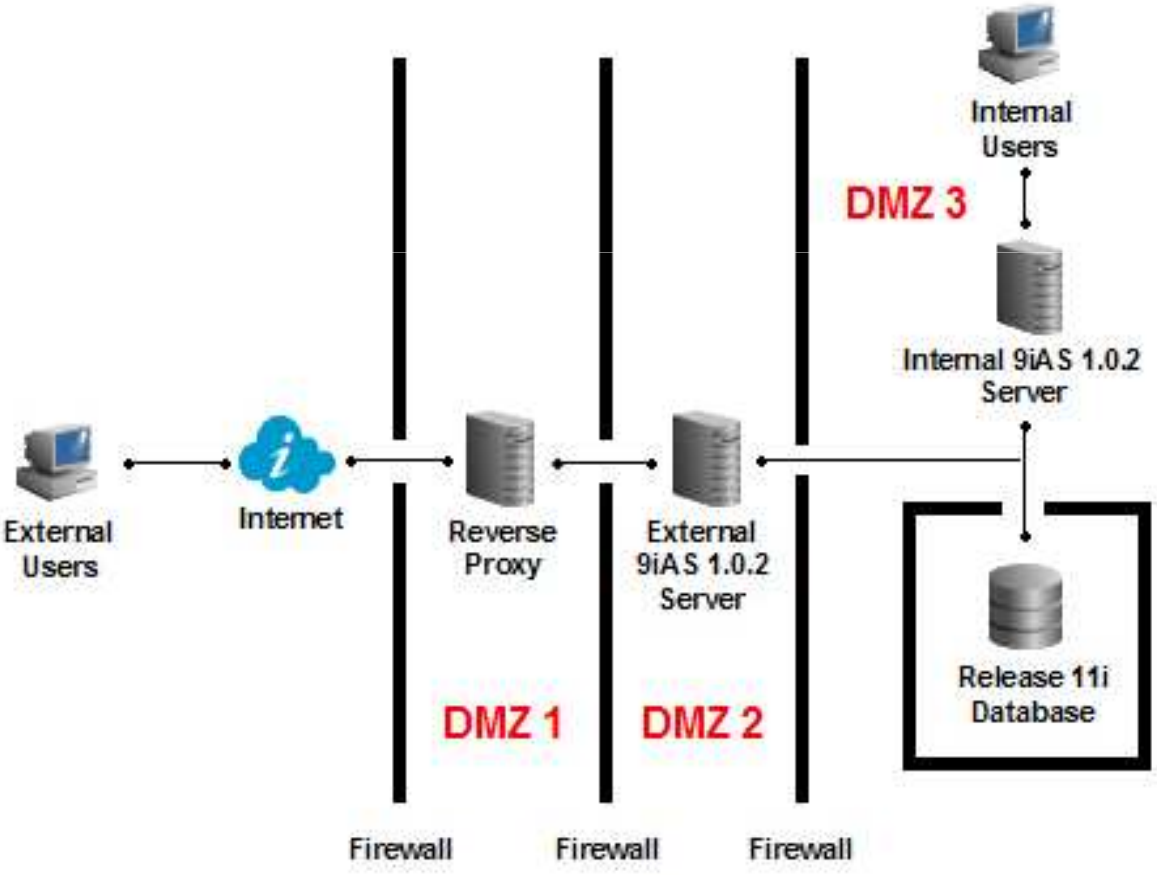
After install it shows Apache 2 with high degree of confidence

After hardening it shows Orion 2 with low degree of confidence



Configure an OHS reverse proxy

Typical 11i DMZ setup. Anything wrong with this picture?



Mod_status vulnerability

- CVE-2007-6388 - Cross-site scripting (XSS) vulnerability in mod_status

  User Enters:

  /server-status?refresh=0;url=http://untrusted-site.com/

  Server Responds with this header:

  Refresh: 0;url=http://untrusted-site.com/

- Refresh parameter entered by the user, and not validated, was placed in an HTTP header

Mod_Security vs Mod_Rewrite

Mod_security

- Pro
  - Availability of Rules
  - Detailed logging
  - Designed as a security tool
- Con
  - New module to maintain
  - Parsing adds overhead
  - OHS uses old 1.84 version

Mod_rewrite

- Pro
  - Typically already in use
  - Good for simple blocking
  - Performance
- Con
  - More work to code rules
  - Logging more for debug
  - Not designed for security

3 - Monitoring the Web Tier

- Grid Control Policies for OHS
- Auditing, Reporting, and Trending (ART)
- Artificial Ignorance (AIg)
- Monster Mitigation Matrix

Grid Control Policies for OHS

DESCRIPTION
Check that HostNameLookup is off on this HTTP Server
Check that MaxKeepAliveRequests directive is set to a non-zero value on this HTTP Server
Verifies that Directory Indexing is disabled
Verifies whether Access Logging is enabled
Verifies that the HTTPd binary is not owned by a super user
Checks whether users other than the owner have write permission in the Document Root folder
Checks whether a Dummy Wallet is being used on HTTP Server
Checks whether Secure Socket Layer (SSL) is enabled for Single Sign-On (SSO) on HTTP Server

Artificial Ignorance (AIg)

- "*Once you eliminate the impossible, whatever remains, no matter how improbable, must be the truth.*" Arthur Conan Doyle
- Gaps in the OHS access log
- Same session cookie from multiple IP addresses
- Requests for content types that you don't serve
- Successful requests for content that you deny

Monster Mitigation Index

ID Name

M1 - Establish and maintain control over all of your inputs.

M2 - Establish and maintain control over all of your outputs.

M3 - Lock down your environment.

M4 - Assume that external components can be subverted, and your code can be read by anyone.

M5 - Use industry-accepted security features instead of inventing your own.

GP1 - Use libraries and frameworks that make it easier to avoid introducing weaknesses.

GP2 - Integrate security into the entire software development lifecycle.

GP3 - Use a broad mix of methods to comprehensively find and prevent weaknesses.

GP4 - Allow locked-down clients to interact with your software.

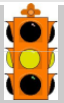
Source: 2010 CWE/SANS Top 25: Monster Mitigations

Ten Most Wanted Characters

WANTED

Session hijacking, Internet flight	*SQL injection, data hijacking*	*Aggravated Execution*	*Web Site Defacement*	*SQL injection*
;	|	`	< >	‘
Sem I. Colon Aliases: %3b	Pi Pe Aliases: %7c	Gra Ves Aliases: %60	L&G Han brothers Aliases: %3c, %3e	Sing LeQuote Aliases: %
SQL injection	*Cybercrime*	*Cache Poisoning*	*Response Splitting*	*Cybercrime*
“	%	\r	\n	~
Doub LeQuote Aliases: %22	Pe R. Cent Aliases: %25	Carri A. Gereturn Aliases: %0d	Lin Efeed Aliases: %0a	Til De Aliases: %7e

4 – Planning for the Future

Configuration Item	Rating
Apache version – recent 2.2.13	
Where is mod_security?	
Conf file macros – good and bad	
Compile options – using only a minimum	
Wildcard include	
Server Signature	
Cgi-bin scripts	
Logging – ODL style	

Tips & Tricks for Managing OHS

Best Feature of OHS 2 not enabled	→Use threads with mpm worker
Build your own moat	→Protect your COTS products
Listen up!	→Make sure you check all ports
Use an inclusive OHS configuration	→Use include to separate configs
Can you use mod_plsql and OHS2	→Yes, and reduce DB connections
Use mod_rewrite or mod_security?	→Why choose, use both
A bit of nostalgia	→New load modules with 2.2
Virtualization	→Inherit rules with Virtualhosts
Load Module order is important	→Load Module order matters in 1.3
Test those changes	→apachectl configtest is OK
Need a little Cache?	→Take advantage of client caching
Terminating SSL in front of OHS	→Speed up your secure requests

Thanks for Attending!

Fill out your comment card – Session # 712

Collaborate10 recommended presentations:

- You vs. The Bad Guys - The Top 10 List For Securing R12
- Securing the E-Business Suite Expert & Best Practices Panel

Contact Information

Brian J. Mulreany

Email: bjm-uva@alumni.virginia.edu

Kevin Sheehan

Email: kevin.sheehan@agilex.com

Blog: http://securedba.com/