How to Secure Oracle WebLogic 12c

Important Security Tips Everyone Should Know

LearnWebLogicOnline.com

- Is your enterprise deployment of WebLogic secure from external or internal threats?
- Do you know exactly what is being exposed to the Internet?
- Do you know what it means to harden a WebLogic installation?
- Do you understand the difference between administering development versus production environments?

Agenda

- Oracle WebLogic Installation
- Domain Security
- Network Security
- Administrative Security

Audience

- Systems and Application Administrators
- Enterprise / Cloud Architects
- Developers
- All of the above with previous WebLogic experience

Other Courses Online

Available at
www.LearnWeblogicOnline.com

- Sign Up for News, Discounts
- Oracle WebLogic 12c for Administrators

Your Instructor…

Chris Parent

Educator / Owner @
LearnWebLogicOnline.com

B.S. Computer Science
M.S. Software Engineering

- Software Development
- Enterprise and Cloud Architectures
- Former BEA/Oracle Middleware Consultant
- Over 15 years in IT

Lecture 1: Production Installation

How to securely prepare and install WebLogic in a Production Environment

Tip #1: Limit # of OS accounts

- More OS accounts = increased security risk
- Recommend using single OS user/group to own install and runtime processes
  - Oracle Home
  - Domain Home
  - Node Manager + JVM instances
- NEVER USE ROOT!

Example

User	: wlsadmin
Group	: oracle

<oracle_home>	wlsadmin:oracle
<domain_home>	wlsadmin:oracle

Tip #2: Remove Development Components

- Do not install sample code, domains, applications
  - Configuration Wizard
  - Derby DB
  - Demo Certificates
  - jCOM tools – Legacy MS COM support

Tip #3: Apply Patches

- Up-to-date patching reduces security risk
  - OS, JDK, Database, WLS, etc…
- Define an enterprise/corporate patching policy
  - What, where, when, how

Oracle Patch Types

- Interim
- Bundle
- Security Patch Update (SPU)
- Patch Set Updates (PSUs) – used to patch WLS only

Where to get Patches

- My Oracle Support: https://support.oracle.com
- Requires OTN account and support ID

Patch Process

1. Download patch(es)
2. Verify patch prerequisites using `opatch apply -report`
3. Apply patch using `opatch apply`
4. Verify patch was applied using `opatch lsinventory`
5. Roll back patch if necessary using `opatch rollback`

OPatch

- Use OPatch
  - ORACLE_HOME/OPatch
  - ./opatch -help
- Smart Update no longer supported in 12c

OPatch Examples

- Apply a single patch
  - `opatch **apply** <location_of_patch>`
- Apply multiple patches
  - `opatch **napply** <location_patch_parent_directory>`
- View/verify applied patches
  - `opatch **lsinventory**`

Summary – Secure Installation

1. Limit # of OS accounts
2. Remove development components
3. Apply patches

Lab #1: Secure Installation

Lecture 2 : Domain Security

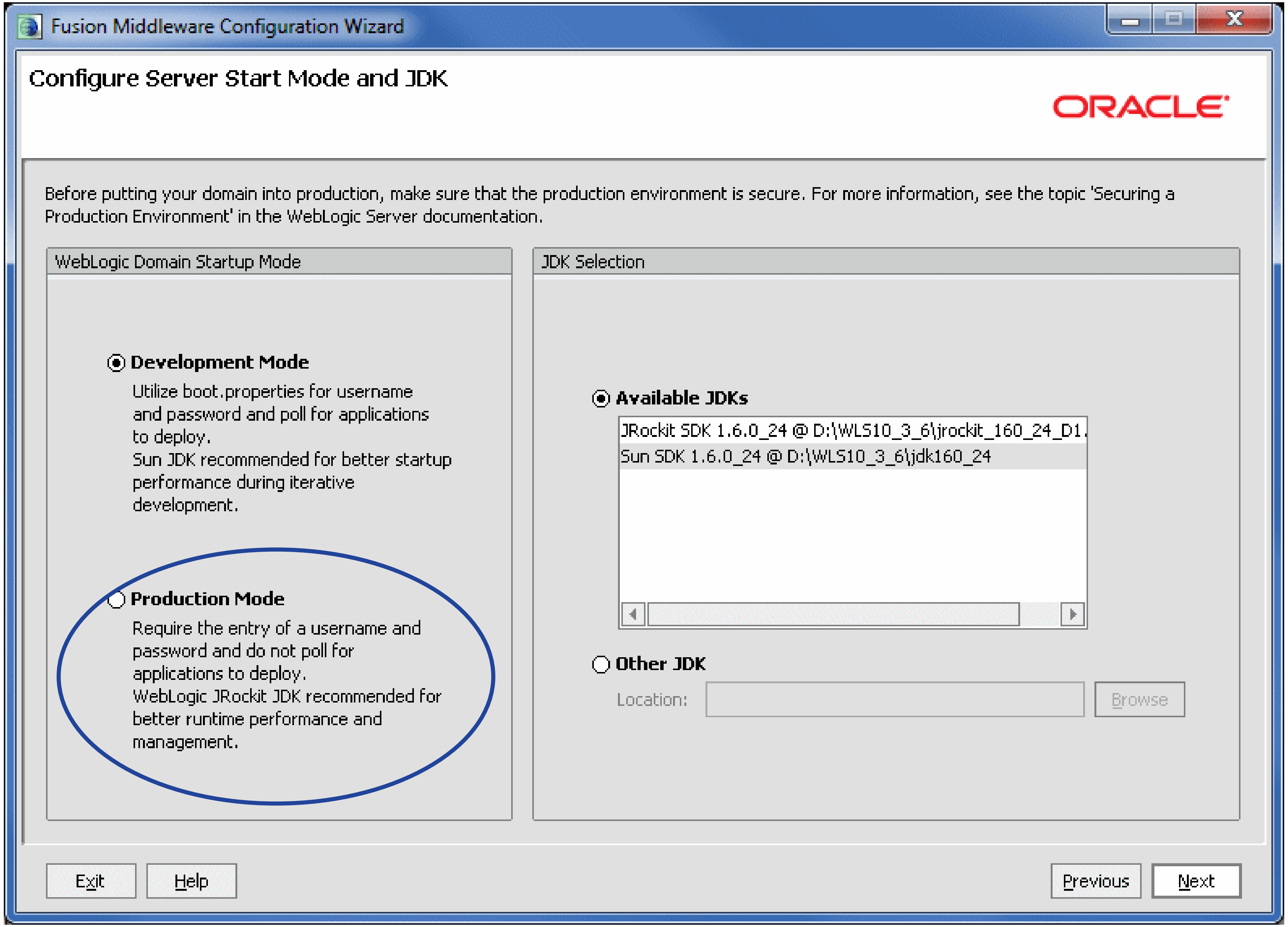
Understand how to protect WebLogic domains

ORACLE
FUSION MIDDLEWARE
WEBLOGIC SERVER

Tip #1 : Use Production Mode

Development	Production
Allows auto-deployment	Auto-deploy disabled
Allows using demo certs	Warning issued if demo certs used
boot.properties created automatically	boot.properties not created
Log files rotated at 500kb	Log files rotated at 5000kb
JDBC capacity – 15 connections	JDBC capacity – 25 connections

ORACLE
FUSION MIDDLEWARE
WEBLOGIC SERVER



Fusion Middleware Configuration Wizard

Configure Server Start Mode and JDK

Before putting your domain into production, make sure that the production environment is secure. For more information, see the topic 'Securing a Production Environment' in the WebLogic Server documentation.

WebLogic Domain Startup Mode

● **Development Mode**

Utilize boot.properties for username and password and poll for applications to deploy.
Sun JDK recommended for better startup performance during iterative development.

○ **Production Mode**

Require the entry of a username and password and do not poll for applications to deploy.
WebLogic JRockit JDK recommended for better runtime performance and management.

JDK Selection

● **Available JDKs**

JRockit SDK 1.6.0_24 @ D:\WLS10_3_6\jrockit_160_24_D1.
Sun SDK 1.6.0_24 @ D:\WLS10_3_6\jdk160_24

○ **Other JDK**

Location:

Tip #2: Create Delegated Admins

- WLS supports roles :
  - Application deployment
  - Resource configuration
  - Monitoring
- Create user accounts and assign to roles using Admin Console – Security Realm

How Passwords are Protected

- Passwords for accessing resources are hashed
- SerializedSystemIni.dat contains hashes
- Associated with a specific domain
- Located in <DOMAIN_HOME>
- Should be backed up
- Only WLS administrator should have rw access

Tip #3 : Enforce Password Policy

- Define and implement password policy using WebLogic's Password Validator
- Domain Security Realm > Security Providers

Tip #4 : Set User Lockout and Limits

- Define user login attempts and account lockout time limits
- Enabled by default

Tip #5 : Enable Security Auditing

- Record key security events
  - AuthN and AuthZ checks
- Implemented using Auditing Provider
  - Define log rotation and severity
- Comes with DefaultAuditor
- Support for custom provider

Configure Default Auditor

- Security Realms > myrealm > Providers > Auditing
- <DOMAIN_HOME>\yourserver\logs\DefaultAuditRecorder.log

Tip #6 : Trusting Domains

- Cross-Domain Security used to trust 2 Domains
- Security principals from one domain can make calls in another domain
- Used by JMS, JTA, MDB, and WAN replication

Configuring Cross Domain Security

- Enable CDS via Admin Console or mbean attr
- Configure Cross-domain user to use CrossDomainConnector role
- Configure Credential Mapper for CDS
- Domain names must be unique

The Old Way of Trusting

- Global Trust is still supported but not recommended
- Trust is established using a single credential
- Trust relationship is transitive and symmetric
  - Domain A = Domain B
  - Domain B = Domain C
  - Domain A = Domain C

Summary – Domain Security

1. Use production mode
2. Create delegated admins
3. Enforce password policies
4. Set user lockout and timeout limits
5. Audit security events
6. Enable trust between domains

Lab #2: Domain Security

Create, configure, and protect a Domain

Lab 2.1 and 2.2 Domain Creation

ORACLE
FUSION MIDDLEWARE
WEBLOGIC SERVER

Lab 2.3 boot.properties

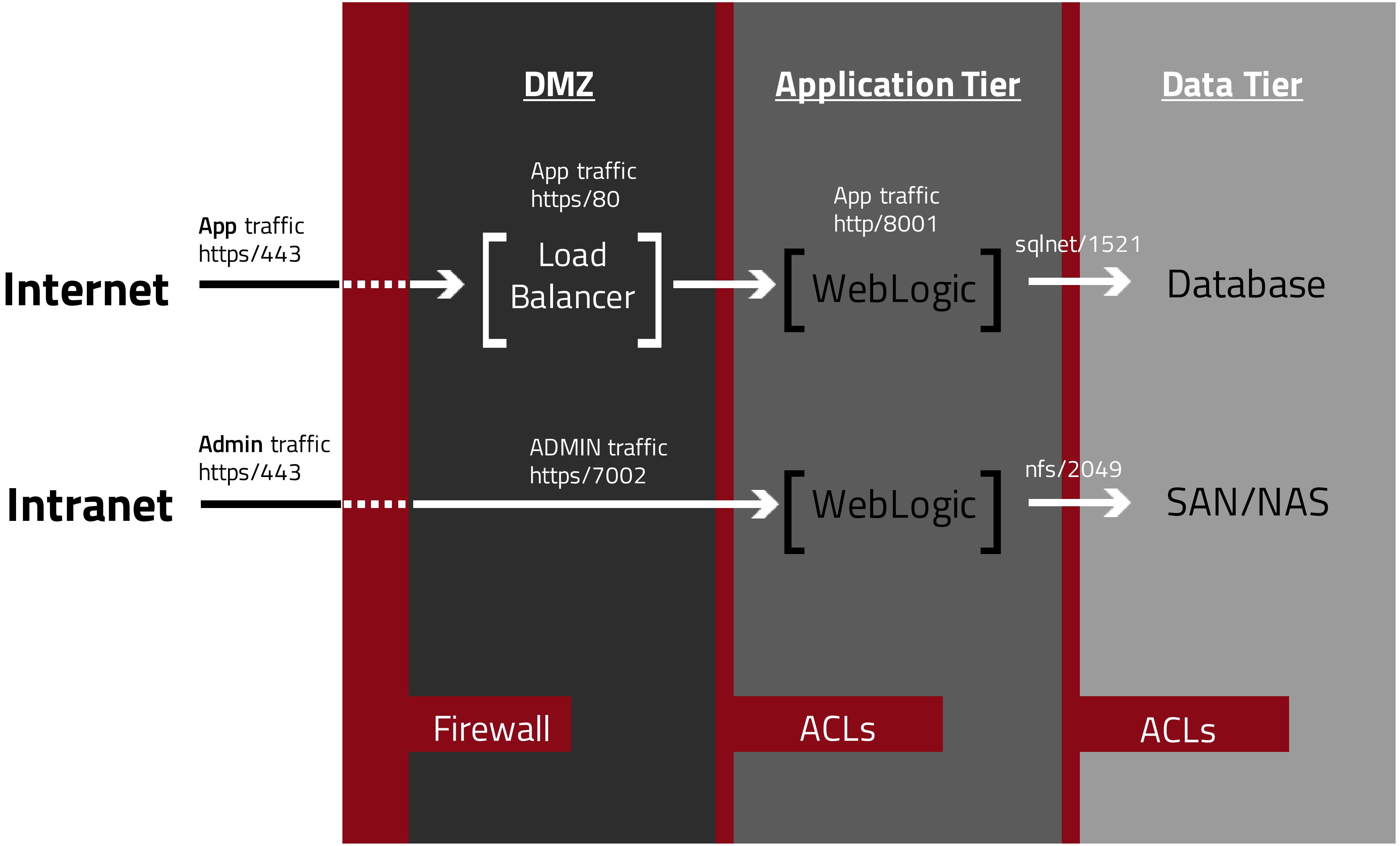
Lab 2.3 boot.properties

Lecture 3 : Network Security

Techniques for securing external and internal access

Tip #1 : Secure Network Architecture

- Use an N-tier deployment architecture to isolate functions
- Use firewalls and ACLs to only expose end-user functions to customers
- Never directly expose WebLogic to the Internet
- End-user and admin functions should be segregated



ORACLE
FUSION MIDDLEWARE
WEBLOGIC SERVER

Default WebLogic Ports

	Default Port	Allotted Port Range
Oracle WebLogic Server Listen Port for Administration Server	7001	7001-9000
Oracle WebLogic Server Listen Port for Managed Server	8001	8000 - 8080
Oracle WebLogic Server Node Manager Port	5556	5556
Oracle WebLogic Server SSL Listen Port for Administration Server	7002	7002-9000

Tip #2: Thwarting DoS Attacks

- Configure Max Message Sizes and Timeouts
- Via the Admin Console
- Per network channel, per protocol, per server

Tip #3 : Use Connection Filters

- Use only when firewall is not available
- Limit traffic based upon:
  - Protocol
  - IP addresses
  - DNS node names
- Used mostly to limit traffic between WLS nodes behind firewall
- Configured using Admin Console

Using a Connection Filter

- Implemented using a Java class and Rules
  - Out of the box: weblogic.security.net.ConnectionFilterImpl

```
127.0.0.1 * 7001 allow #local ipv4
0:0:0:0:0:0:0:1 * 7001 allow #local ipv6
0.0.0.0/0 * 7001 deny # all other traffic
```

Lab #3: Network Security

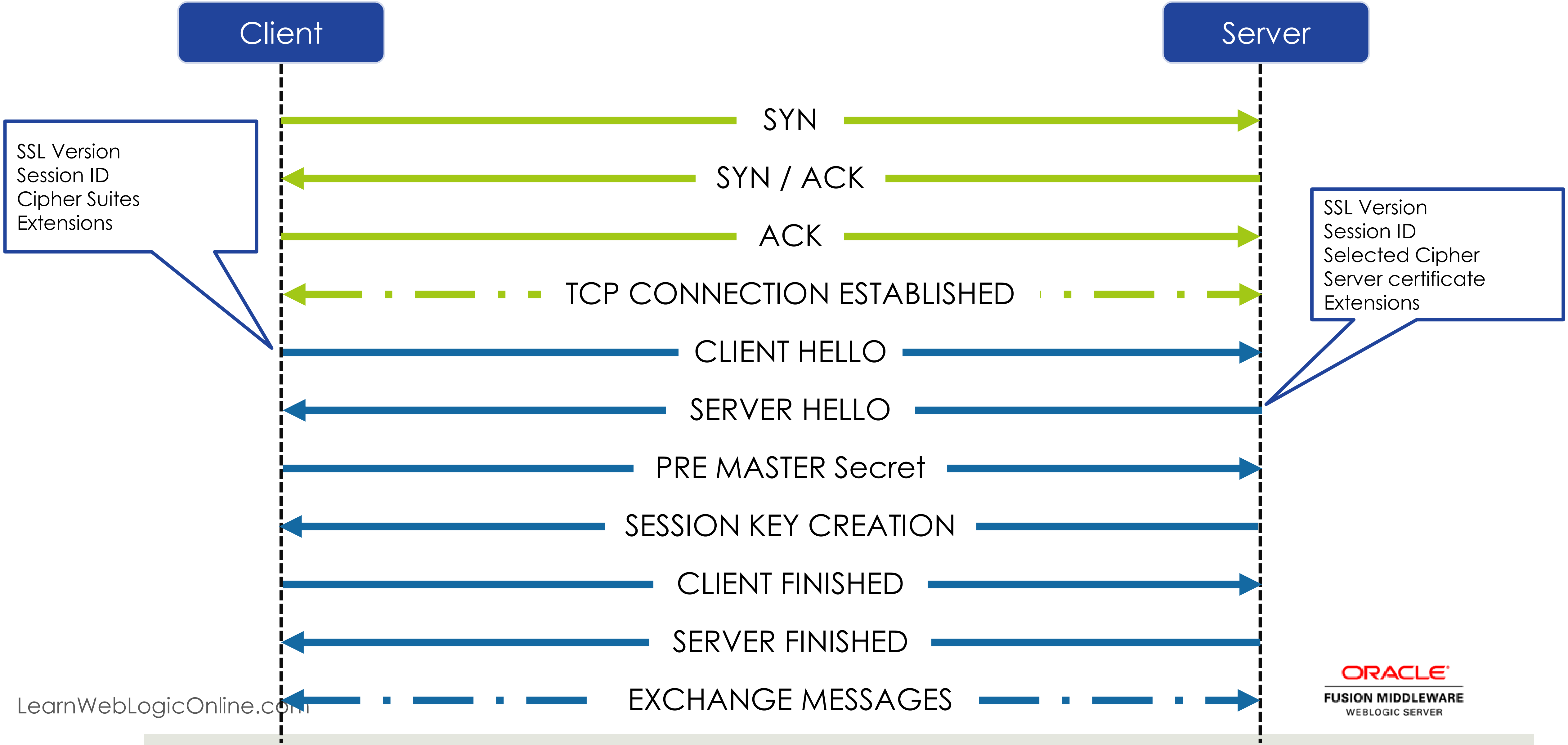
Tip #4 : Encrypt Traffic using SSL

- Use SSL/TLS to encrypt network traffic
- Used to protect application and/or administrative traffic
- Requires creating digital certs and configuring Identity and Trust

SSL / TLS

- SSL uses public key encryption for authN
- Public and Private key generated for server
- Public key embedded in digital certificate
- Data encrypted with public key
- Decrypted with private key
- Third-part or CA validates public key – establishes trust

SSL/TLS Handshake



Client

Server

SSL Version
Session ID
Cipher Suites
Extensions

SYN

SYN / ACK

ACK

TCP CONNECTION ESTABLISHED

CLIENT HELLO

SERVER HELLO

SSL Version
Session ID
Selected Cipher
Server certificate
Extensions

PRE MASTER Secret

SESSION KEY CREATION

CLIENT FINISHED

SERVER FINISHED

EXCHANGE MESSAGES

Identity and Trust

- **Identity** = Private key + Digital certificate
- **Trust** = Trusted CA certificate(s)
- Keys and certificates stored in Keystores (JKS, JKCS)
- Configured for each server acting as an SSL client/server

Private Key

- WebLogic uses **Public Key Encryption** for authentication
- Public and private keys generated for each server
- Data **encrypted** with **Public Key**
- Data **decrypted** with **Private Key**

Digital Certificate

- Electronic document used to verify identity of an entity
- Binds identity of user or entity to a public key
- Verified by a trusted third party (trusted CA)
- Most common format x.509

Certificate Authority

- Issues digital certificates
- Signs digital certificate with its own private key
- Digital certificate verified by using CA’s public key

Configuring Identity and Trust

1. Obtain public, private keys and digital certificate for each server
2. Create Identity and Trust keystores
3. Store public, private keys and digital certs in keystores
4. Configure keystores for each WLS server

Configuring SSL

1. Configure Identity and Trust from previous slide
2. Set SSL configuration options for private key alias and password
3. Enable or disable host name verification
4. Enable SSL listen port

SSL Best Practices

- Enable TLS1.0 or greater
- Enable Host Name Verification
- Self-signed certificates OK for Internal Use Only
- Always create strong server certs
- Disable weak CIPHERS
- **NEVER USE DEMO CERTS!**

Debugging SSL

- Getting SSL to work can be tricky
- Enable SSL debug flags at the JVM level
- Debug trace displays:
  - SSL server config info
  - Trusted Cas
  - Server identity
  - Encryption strength allowed
  - Enabled ciphers
  - SSL handshake

JVM Arguments

```
-Djavax.net.debug=all
-Dssl.debug=true
-Dweblogic.StdoutDebugEnabled=true
```

Summary – Network Security

1. Create a secure network architecture
2. Set message limits to prevent DOS attacks
3. Use connection filters
4. Encrypt traffic using SSL

Lab #4: Encrypting Traffic

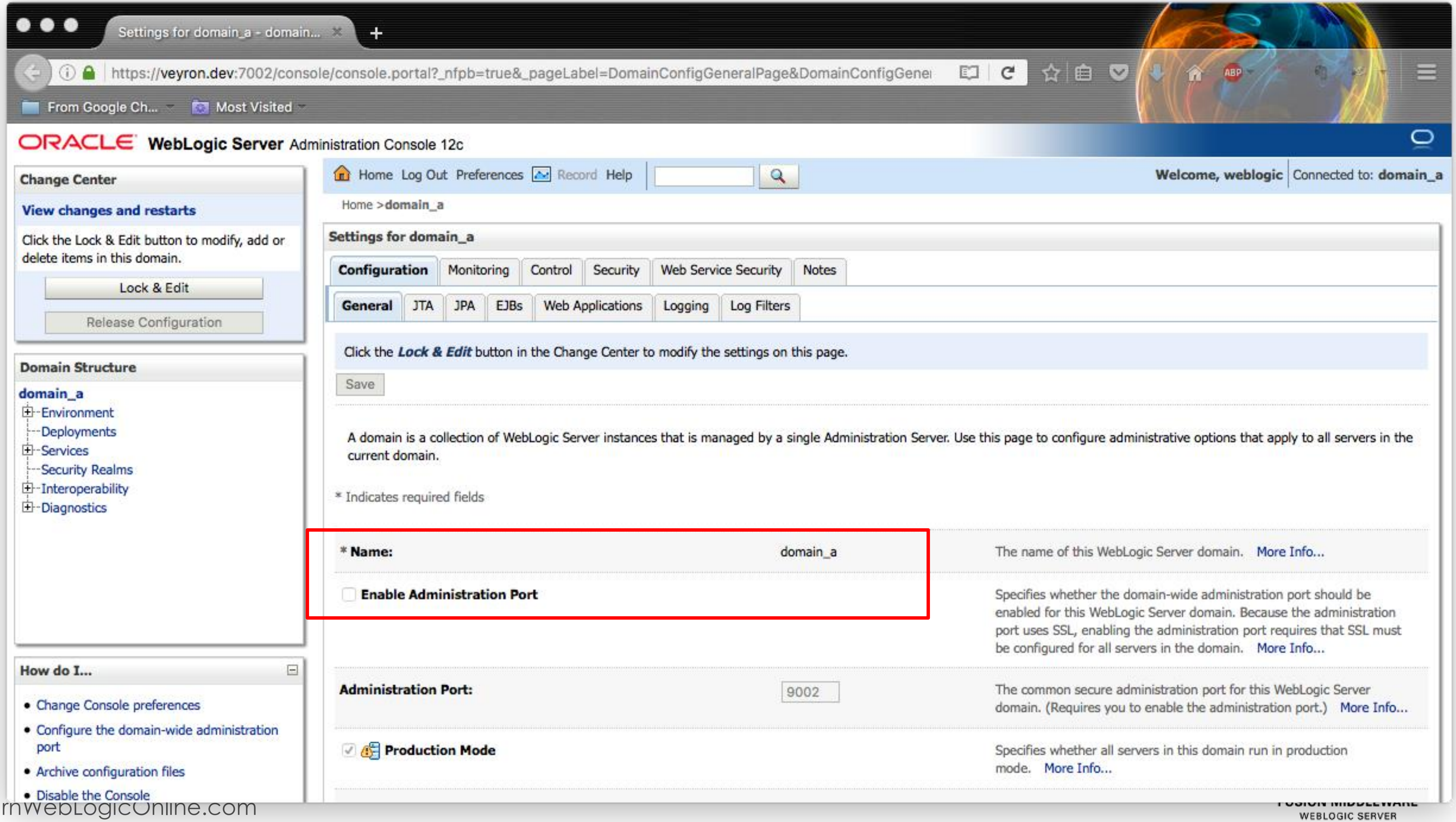
Lecture 4 : Administrative Security

How to securely administer WebLogic

ORACLE
FUSION MIDDLEWARE
WEBLOGIC SERVER

Tip #1: Use Administration Port

- Segregates admin traffic to dedicated network channel
- Runs on separate thread
- Requires additional listen port for each server
- Requires SSL
- Impacts all WLS servers, NodeManager, and any WLST scripts



Settings for domain_a - domain...
https://veyron.dev:7002/console/console.portal?_nfpb=true&_pageLabel=DomainConfigGeneralPage&DomainConfigGene
From Google Ch...
Most Visited
ORACLE WebLogic Server Administration Console 12c
Home Log Out Preferences Record Help
Welcome, weblogic Connected to: domain_a

Change Center

View changes and restarts

Click the Lock & Edit button to modify, add or delete items in this domain.

Lock & Edit

Release Configuration

Domain Structure

domain_a
- Environment
- Deployments
- Services
- Security Realms
- Interoperability
- Diagnostics

How do I...

- Change Console preferences
- Configure the domain-wide administration port
- Archive configuration files
- Disable the Console

Home >**domain_a**

Settings for domain_a

Configuration | Monitoring | Control | Security | Web Service Security | Notes

General | JTA | JPA | EJBs | Web Applications | Logging | Log Filters

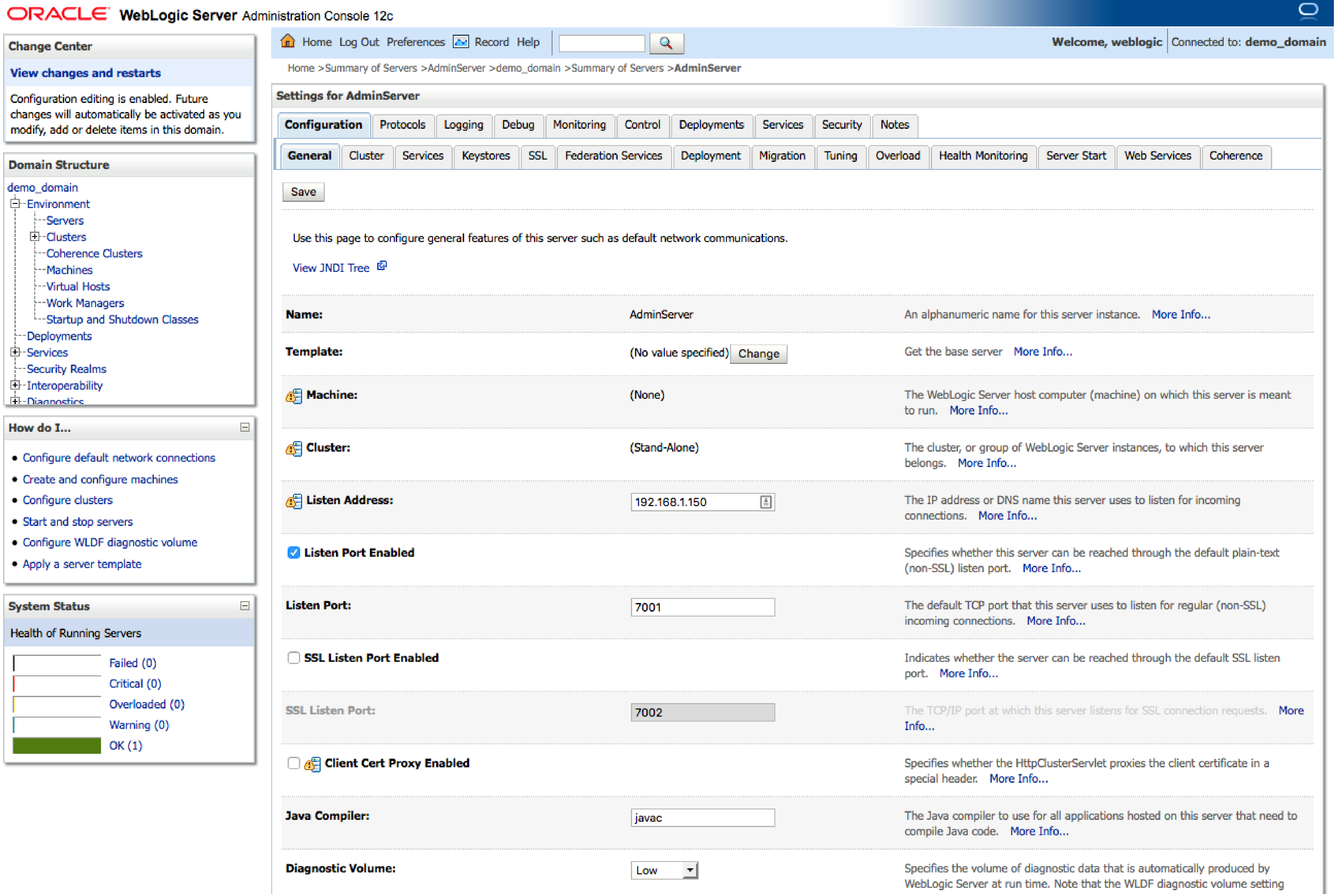
Click the ***Lock & Edit*** button in the Change Center to modify the settings on this page.

Save

A domain is a collection of WebLogic Server instances that is managed by a single Administration Server. Use this page to configure administrative options that apply to all servers in the current domain.

* Indicates required fields

* **Name:**	domain_a	The name of this WebLogic Server domain. More Info...
□ **Enable Administration Port**		Specifies whether the domain-wide administration port should be enabled for this WebLogic Server domain. Because the administration port uses SSL, enabling the administration port requires that SSL must be configured for all servers in the domain. More Info...
Administration Port:	9002	The common secure administration port for this WebLogic Server domain. (Requires you to enable the administration port.) More Info...
☑ **Production Mode**		Specifies whether all servers in this domain run in production mode. More Info...



ORACLE WebLogic Server Administration Console 12c
Home Log Out Preferences Record Help
Welcome, weblogic Connected to: demo_domain
Home >Summary of Servers >AdminServer >demo_domain >Summary of Servers >AdminServer
Configuration Protocols Logging Debug Monitoring Control Deployments Services Security Notes
General Cluster Services Keystores SSL Federation Services Deployment Migration Tuning Overload Health Monitoring Server Start Web Services Coherence
Save

Change Center

View changes and restarts

Configuration editing is enabled. Future changes will automatically be activated as you modify, add or delete items in this domain.

Domain Structure

- demo_domain
  - Environment
    - Servers
    - Clusters
    - Coherence Clusters
    - Machines
    - Virtual Hosts
    - Work Managers
    - Startup and Shutdown Classes
  - Deployments
  - Services
  - Security Realms
  - Interoperability
  - Diagnostics

How do I...

- Configure default network connections
- Create and configure machines
- Configure clusters
- Start and stop servers
- Configure WLDF diagnostic volume
- Apply a server template

System Status

Health of Running Servers

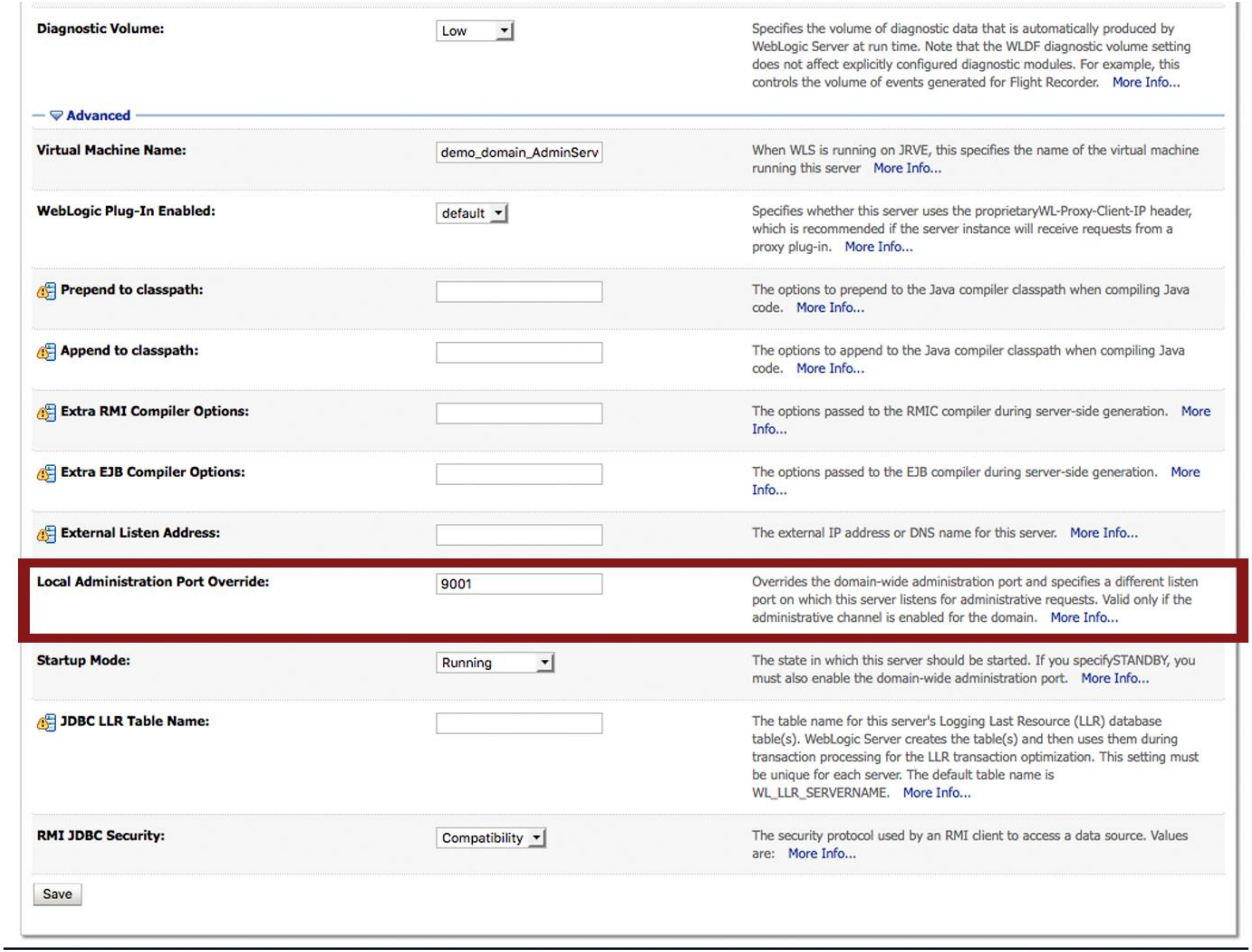
Failed (0)
Critical (0)
Overloaded (0)
Warning (0)
OK (1)

Settings for AdminServer

Use this page to configure general features of this server such as default network communications.

View JNDI Tree

Name:	AdminServer	An alphanumeric name for this server instance. More Info...
Template:	(No value specified) Change	Get the base server More Info...
Machine:	(None)	The WebLogic Server host computer (machine) on which this server is meant to run. More Info...
Cluster:	(Stand-Alone)	The cluster, or group of WebLogic Server instances, to which this server belongs. More Info...
Listen Address:	192.168.1.150	The IP address or DNS name this server uses to listen for incoming connections. More Info...
☑ **Listen Port Enabled**		Specifies whether this server can be reached through the default plain-text (non-SSL) listen port. More Info...
Listen Port:	7001	The default TCP port that this server uses to listen for regular (non-SSL) incoming connections. More Info...
□ **SSL Listen Port Enabled**		Indicates whether the server can be reached through the default SSL listen port. More Info...
SSL Listen Port:	7002	The TCP/IP port at which this server listens for SSL connection requests. More Info...
□ **Client Cert Proxy Enabled**		Specifies whether the HttpClusterServlet proxies the client certificate in a special header. More Info...
Java Compiler:	javac	The Java compiler to use for all applications hosted on this server that need to compile Java code. More Info...
Diagnostic Volume:	Low	Specifies the volume of diagnostic data that is automatically produced by WebLogic Server at run time. Note that the WLDF diagnostic volume setting



Diagnostic Volume:	Low	Specifies the volume of diagnostic data that is automatically produced by WebLogic Server at run time. Note that the WLDF diagnostic volume setting does not affect explicitly configured diagnostic modules. For example, this controls the volume of events generated for Flight Recorder. More Info...

Advanced

Virtual Machine Name:	demo_domain_AdminServ	When WLS is running on JRVE, this specifies the name of the virtual machine running this server More Info...
WebLogic Plug-In Enabled:	default	Specifies whether this server uses the proprietaryWL-Proxy-Client-IP header, which is recommended if the server instance will receive requests from a proxy plug-in. More Info...
Prepend to classpath:		The options to prepend to the Java compiler classpath when compiling Java code. More Info...
Append to classpath:		The options to append to the Java compiler classpath when compiling Java code. More Info...
Extra RMI Compiler Options:		The options passed to the RMIC compiler during server-side generation. More Info...
Extra EJB Compiler Options:		The options passed to the EJB compiler during server-side generation. More Info...
External Listen Address:		The external IP address or DNS name for this server. More Info...
Local Administration Port Override:	9001	Overrides the domain-wide administration port and specifies a different listen port on which this server listens for administrative requests. Valid only if the administrative channel is enabled for the domain. More Info...
Startup Mode:	Running	The state in which this server should be started. If you specifySTANDBY, you must also enable the domain-wide administration port. More Info...
JDBC LLR Table Name:		The table name for this server's Logging Last Resource (LLR) database table(s). WebLogic Server creates the table(s) and then uses them during transaction processing for the LLR transaction optimization. This setting must be unique for each server. The default table name is WL_LLR_SERVERNAME. More Info...
RMI JDBC Security:	Compatibility	The security protocol used by an RMI client to access a data source. Values are: More Info...

Tip #2: Avoid Plain Text Passwords

- Never hardcode passwords in scripts
- Never enter passwords in command line
- Passwords will show up in process listings, shell history, log files, etc

Password Solution

- Use a Key File for AuthN
- Contains encrypted username and password
- Use WLST command **storeUserConfig**() to generate key file
- Specify key file as parameter when connecting using WLST

User Configuration File Example

- Create user config and key file – Must be connected to running server
  - wls:/demodomain/serverConfig>storeUserConfig('/usr/home/user1/configfile.secure', '/usr/home/user1/keyfile.secure')
- Connect to Weblogic using user config
  - wls:/offline> connect(userConfigFile='/usr/home/user1/configfile.secure', userKeyFile='/usr/home/user1/keyfile.secure', url='t3://host:port')

Summary – Administrative Security

1. Use administration port/channel

2. Use user config files and keys for WLST AuthN

Conclusion

Let's recap what we have learned.

What We’ve Learned

- Secure installation
- Domain Security
- Network Security
- Administrative Security

What's Next?

- Application security
- Identity Management – OIM, OAM
- Oracle HTTP Server / Apache and Webgates
- Directory Services – LDAP, OUD/OID

THE END?

- Questions?
- Comments?
- Improvements?
- Additional topics?

chris@LearnWeblogicOnline.com

www.LearnWeblogicOnline.com

Twitter @learnweblogic

THANK YOU!