SQL Server Security

Chad Boyd

chad.boyd.tips@gmail.com

http://blogs.mssqltips.com/blogs/chadboyd

http://www.mssqltips.com

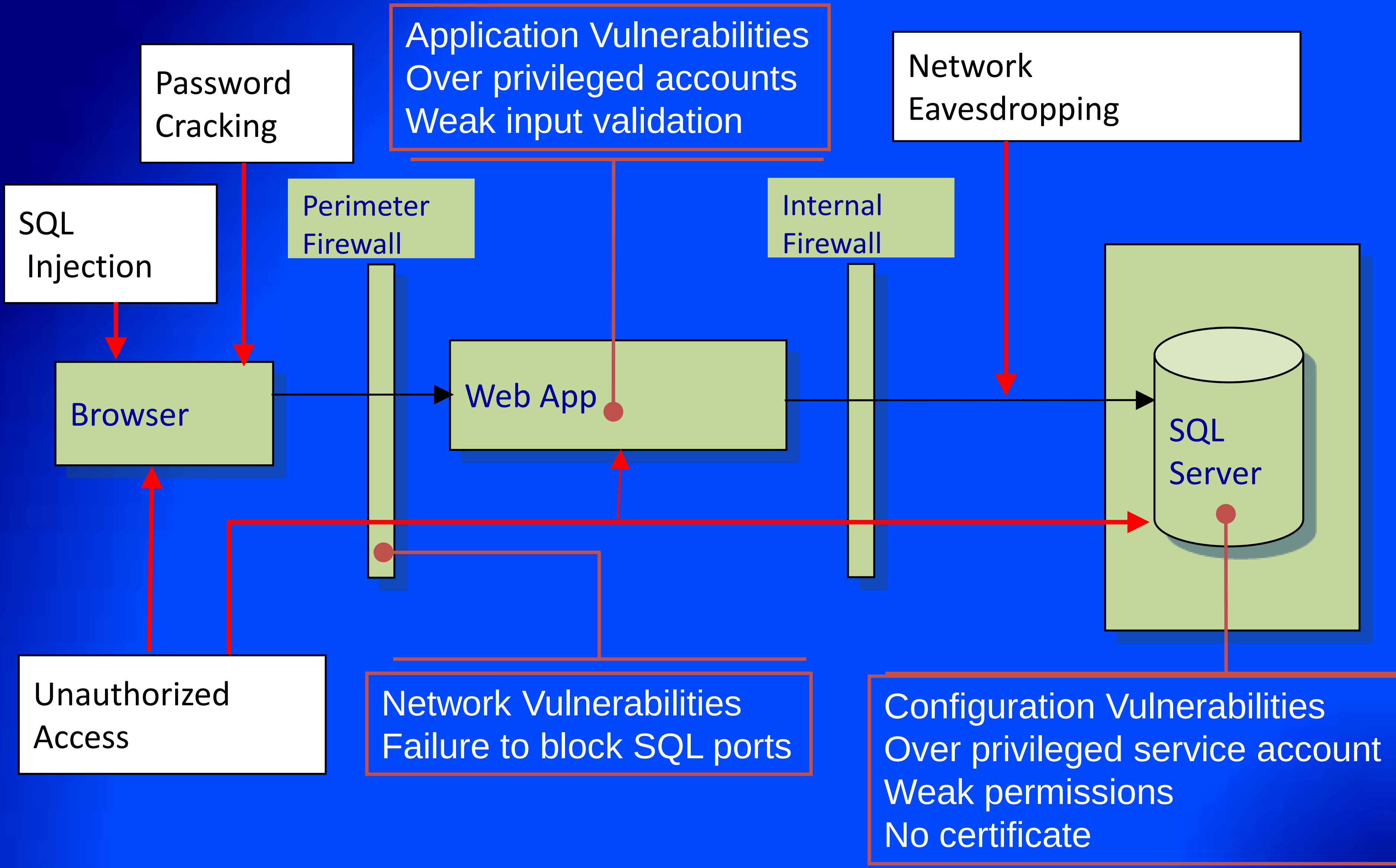
Agenda

- Introduction
- SQL Security Best Practices
- Looking ahead – SQL 2008

Agenda

- Introduction
- SQL Security Best Practices
- Looking ahead – SQL 2008

Threats and Vulnerabilities



Password Cracking

Application Vulnerabilities
Over privileged accounts
Weak input validation

Network Eavesdropping

SQL Injection

Perimeter Firewall

Internal Firewall

Browser

Web App

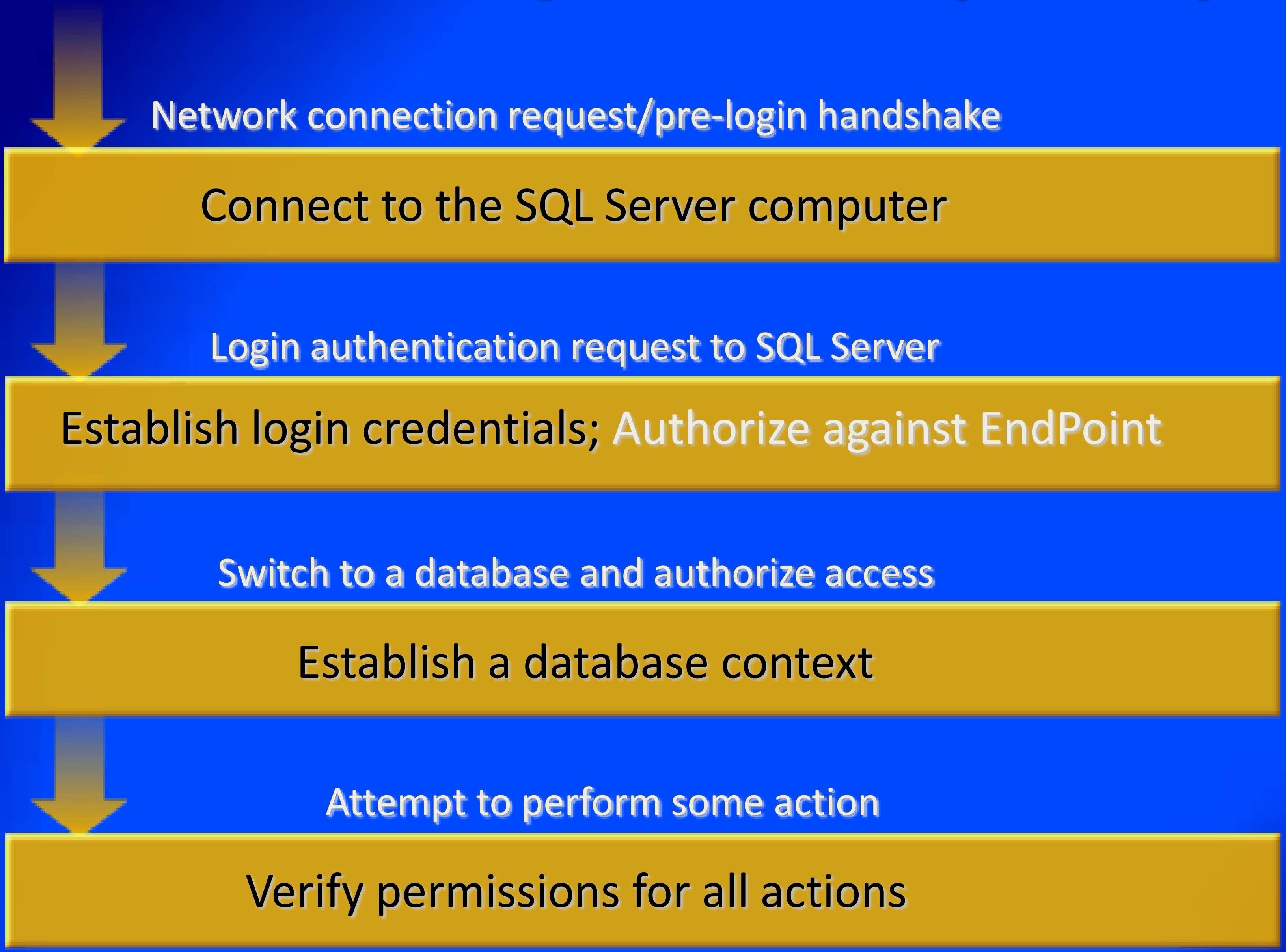
SQL Server

Unauthorized Access

Network Vulnerabilities
Failure to block SQL ports

Configuration Vulnerabilities
Over privileged service account
Weak permissions
No certificate

SQL Security Model (2005)



Network connection request/pre-login handshake

Connect to the SQL Server computer

Login authentication request to SQL Server

Establish login credentials; Authorize against EndPoint

Switch to a database and authorize access

Establish a database context

Attempt to perform some action

Verify permissions for all actions

Agenda

- Introduction
- SQL Security Best Practices
- Looking ahead – SQL 2008

1) Surface Area Reduction

- What
  - Minimize Enabled/Exposed features
  - "Off by Default" for new SQL2K5 installs
  - Features, services, connections
- Recommendations
  - Enable only what you will actually use
  - Keep connectivity to a minimum
  - Upgrades - turn off whatever you don’t need
- Why
  - Reduced attack surface
  - Heterogeneous installation footprint
- How
  - Surface Area Config., Config. Manager
  - sp_configure / TSQL

2) Service Accounts

- What
  - Services can run under built-in account, local/domain user account
  - Each service can use a different account
- Recommendations
  - Most desirable - Local or Domain account
  - Least desirable – local system
  - Use different accounts for different services
- Why
  - Least Privilege
  - Isolation
  - Defense in depth
- How
  - Specify at installation
  - SQL Configuration Manager

3) Authentication Mode

- What
  - Windows Authentication Mode
  - Mixed Authentication Mode
- Recommendations
  - Windows Authentication whenever possible
  - Use Mixed Authentication to get
    - Legacy application support
    - Cross platform client/server
    - Improved administrator separation
  - Encrypt communications channel
- Why
  - Single sign on
  - Simplified administration
  - No password management
  - Protect conversations and credentials in transit
- How
  - Installation and SSMS

4) Network Connectivity

- What
  - Protocols and endpoints enabled
  - Demands on strength of channel protection
- Recommendations
  - Enable minimal protocols (e.g. TCP/IP)
  - Change and block default ports (1433, 1434)
  - Grant user access through restrictive endpoints
  - Do not expose to internet
- Why
  - Minimize potential client population
  - Block known attacks
  - Restrict access paths
- How
  - Surface Area Configurator
  - SQL Configuration Manager
  - TSQL / ENDPOINT DDL

5) System Procedures (xp's)

- What
  - xp_cmdshell, xp_regread, xp_dirtree, etc.
  - sp_OA*
- Recommendations
  - Limit usage and authorized users
  - 2005 – turn off if not used
  - DO NOT remove (unsupported configuration)
- Why
  - Improper usage can lead to escalated priveledges
  - Many are off by default in 2005
  - Many contain appropriate authorization check
- How
  - Surface Area Configurator
  - SSMS
  - TSQL

6) Password Policy

- What
  - Complexity, Expiration, Lockout enforcement
    - Common across Windows and SQL
    - Win2K3 onwards (hard-coded rules for older versions)
  - SQL Logins, App Roles, pass phrases, etc.
    - Everywhere passwords are used
- Recommendations
  - Leave CHECK_POLICY on
  - Set CHECK_EXPIRATION on to avoid old passwords
  - Set MUST_CHANGE for new logins
- Why
  - Deter brute-force and dictionary attacks
  - Prevent blank passwords
- How
  - TSQL / SSMS for SQL Logins
  - Domain/machine settings for Windows-based

7) Admin Privileges

- What
  - Principals with highly elevated privileges
  - “sa” built-in login
  - Members of SYSADMIN built-in server role
  - Holders of CONTROL permission at server level
- Recommendations
  - Use admin privileges only when needed
  - Minimize number of administrators
  - Provision admin principals explicitly
  - Have multiple distinct admins if more than one needed
  - Avoid dependency on builtin\administrators Windows group
- Why
  - Least privilege
  - Repudiation/accountability
  - Limit administrative rights into IT
- How
  - SSMS
  - TSQL

8) Database Ownership and Trust

- What
  - Each database is owned by
    - DBO user (default = database creator)
    - DB_OWNER role members
  - Can confer trust on other databases
- Recommendations
  - Have distinct owners for databases
    - Not all owned by "sa"
    - Minimize owners for each database
  - Confer trust selectively
  - Leave CDOC setting off
    - Migrate usage to selective trust instead
- Why
  - Least privilege
  - Repudiation/accountability
  - Isolation
- How
  - Alter authorization on database
  - Trustworthy setting / Signed Modules

9) Schemas

- What
  - Namespace in the container hierarchy
    - Server>database->schema->object
  - Can be owned by any user (SQL2K5)
  - Permissions grantable at schema level
- Recommendations
  - Group related objects together into same schema
  - Leverage ownership and permissions at schema level
  - Have distinct owners for schemas
    - Not all owned by "dbo"
    - Minimize owners for each schema
- Why
  - Isolation, aggregation
  - Flexibility
    - Separate administrative grouping from application access
    - Change owner without updating applications
    - Authorization level
- How
  - TSQL, SSMS

10) Authorization

- What
  - Who can access what
- Recommendations
  - Encapsulate access within modules
  - Manage permissions via database roles
  - Leverage permission granularity
    - Many new permissions in SQL 2005
  - Do not enable Guest access
  - Use Login-less users instead of Application Roles
- Why
  - Least Privilege
  - Administrative ease
  - Avoid password management
- How
  - TSQL, SSMS

11) Execution Context

- What
  - SQL context in which statements execute
  - Explicitly set at execution time
  - Implicitly set when entering module
  - Contexts stack and can be reverted
- Recommendations
  - Consider setting context on modules
  - Use EXECUTE AS instead of SETUSER
  - Use WITH NO REVERT/COOKIE instead of App Roles
- Why
  - Object encapsulation
  - Controlled privilege escalation
- How
  - EXECUTE AS clause
  - WITH NOREVERT…
  - WITH COOKIE INTO…

12) Linked/Remote Servers

- What
  - Enable access to OLEDB data sources on remote servers
  - Remote Servers are deprecated
  - Linked Servers supersede Remote Servers
- Recommendations
  - Phase out any Remote Server Definitions
    - Replace with Linked Servers
- Why
  - Remote Server shortcomings
    - Forces source server to handle passwords, or
    - Disable password checking on target server with "trusted" option
  - Linked Servers authentication
    - Support Windows logins and delegation
    - Protection for SQL logins
      - Encrypt password as part of definition
      - Password protected in transit using self-signed certificate
- How
  - TSQL
  - SSMS

13) Encryption

- What
  - Cryptographic protection of data at rest
  - Applicable at column and cell level
  - Algorithm choices depends on operating system
- Recommendations
  - Encryption is very scenario specific
  - Encrypt high value/sensitive data
    - Symmetric key for data, asymmetric key to protect symmetric key
    - Password protect keys and remove master key encryption for most secure configuration
- Why
  - Protection of data at rest (lost laptop, backups)
  - Advanced/selective access control
  - Need permission AND key to see data
- How
  - TSQL key generation
  - TSQL functions to encrypt/decrypt in variety of ways

14) Auditing

- What
  - Record of security relevant activity
  - Profile system and track potential security violations
- Recommendations
  - Auditing is very scenario specific
  - Password policy in place -> audit failed logins
  - Sensitive database content -> audit security events
    - Including successful logins
  - Increase the default # of error logs rotated
- Why
  - Profile system and track potential security violations
  - Forensic analysis of incidents
- How
  - SQL Trace
  - Server Configuration (out to ERRORLOG)
  - C2 Auditing (if completely necessary)

15) Patching

- What
  - Keeping software up to date with security fixes
  - SQL2000 SP4 onwards:
    - Patching via Microsoft Update
  - SQL2005 onwards:
    - Separate code line for security fixes
- Recommendations
  - Stay as current as possible!
  - Enable automatic updates (where appropriate)
- Why
  - Old attacks never go away
  - Proliferation of installations
  - New issues can occur at any time
- How
  - Enable automatic updates, or
  - Run Microsoft Update explicitly

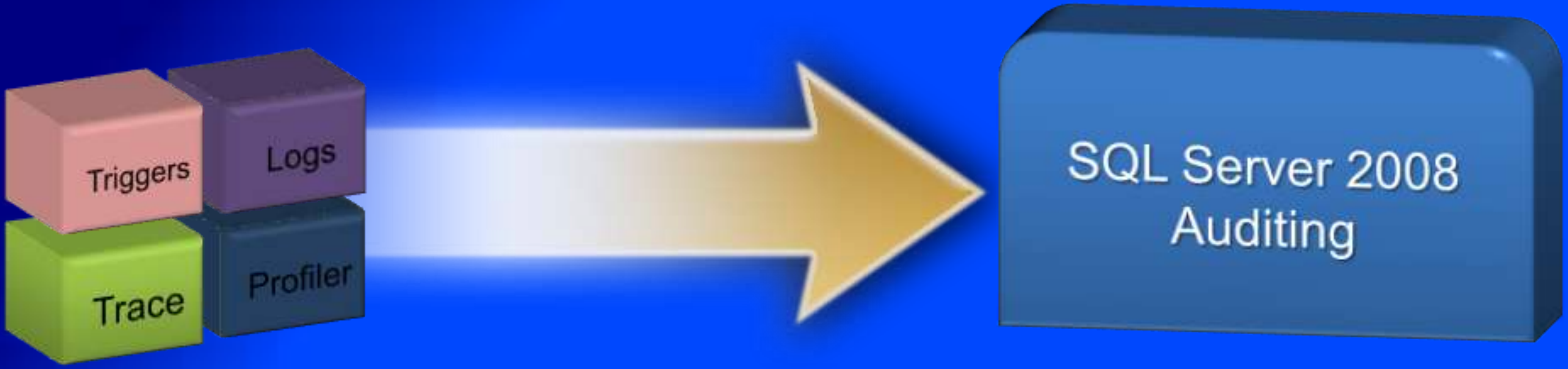
Agenda

- Data Security Landscape
- SQL Security Best Practices
- Looking ahead – SQL 2008

Transparent Data Encryption

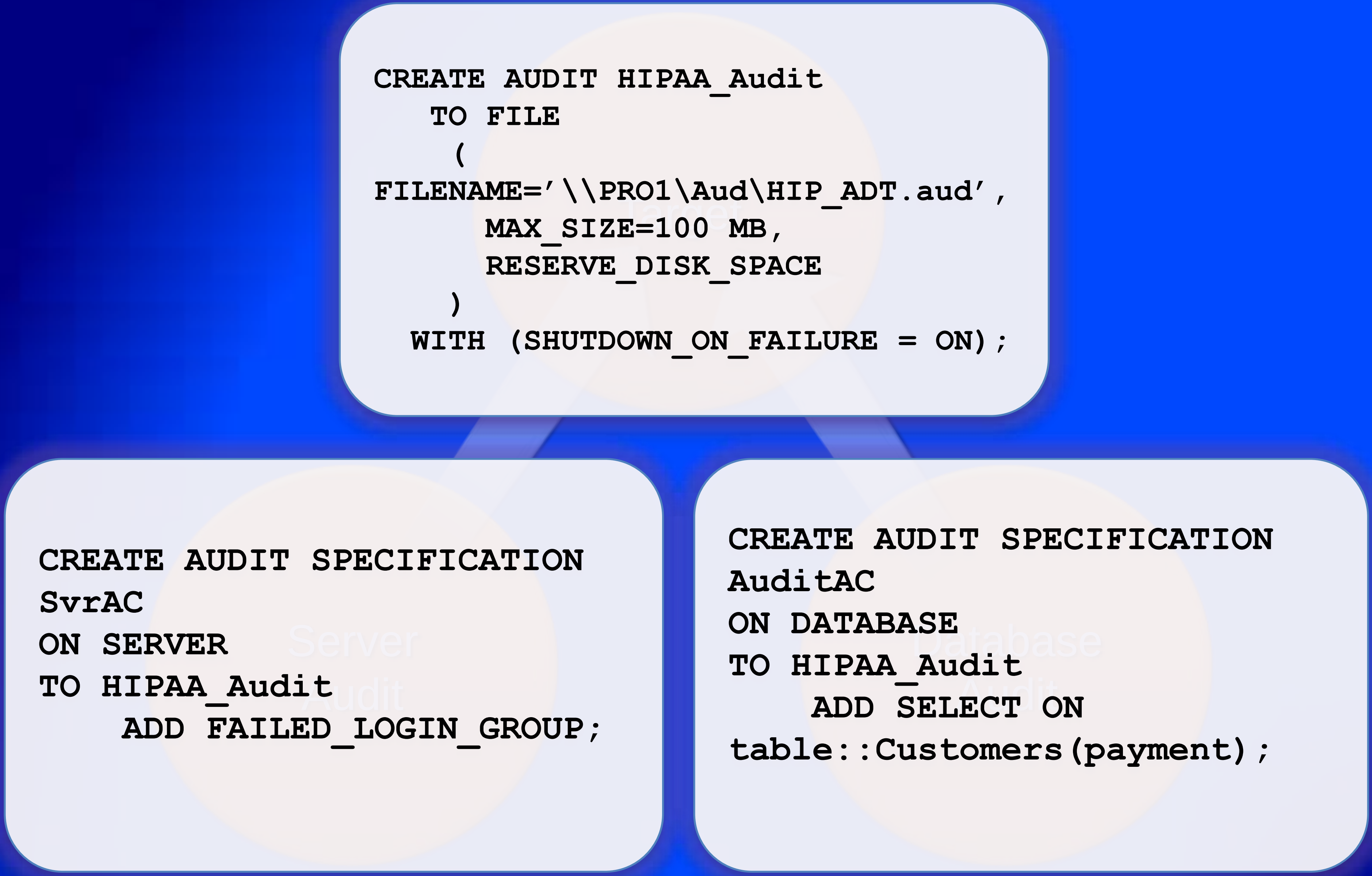
- Encryption with application transparency
  - Expands the SQL Server 2005 encryption offering
- Database level scope
  - Introduces Database Encryption Key (DEK)
- Data at rest protection
  - Decrypted / Encrypted as data moves from / to disk and cache
- Backups are encrypted optionally
- External Key Management
  - Consolidate security keys in the data center

Auditing



- AUDIT is a first Class Server Object
- Granular audit on objects and/or users
- Multiple outputs (File, Windows Application Log, Security Event Log)
- High Performance - based on Extended Events
- Built in tools for consolidation, support for AS/RS

Auditing



```
CREATE AUDIT HIPAA_Audit
   TO FILE
    (
FILENAME=’\\PRO1\Aud\HIP_ADT.aud’,
      MAX_SIZE=100 MB,
      RESERVE_DISK_SPACE
    )
  WITH (SHUTDOWN_ON_FAILURE = ON);
```

```
CREATE AUDIT SPECIFICATION
SvrAC
ON SERVER
TO HIPAA_Audit
    ADD FAILED_LOGIN_GROUP;
```

```
CREATE AUDIT SPECIFICATION
AuditAC
ON DATABASE
TO HIPAA_Audit
    ADD SELECT ON
table::Customers(payment);
```

Policy Based Management

- Spend less time on ongoing operations
- Manage via policies instead of scripts
- Define Enterprise wide data management policies
- Automated monitoring and enforcement of policies
- Simplify your installation and configuration
- Integrated with your enterprise system management
- Define Policies that are compliant with System Definition Model
- Manage your data and system infrastructure with Microsoft System Center

Questions?

Chad Boyd
chad.boyd.tips@gmail.com
http://blogs.mssqltips.com/blogs/chadboyd
http://www.mssqltips.com

http://www.microsoft.com/sql/technologies/security/default.mspx

http://blogs.msdn.com/lcris

http://blogs.msdn.com/sqlsecurity/

http://blogs.msdn.com/raulga/

http://www.sqlsecurity.com/

TITLE

- Test