RSA® Authentication Manager Administration

Rev A0
(ED AMADM310)

RSA EMC²

Contact Information
Go to the RSA corporate website for regional Customer Support telephone and fax numbers: www.emc.com/domains/rsa/index.htm.

RSA Authentication Manager Administration

Table of Contents

Introduction

- Scope of the course
- Personal Introductions
- Logistics
- Expectations
- Schedule
- Lab Environment

RSA EMC²

Agenda

- Product & Technology Overview
- RSA SecurID Structure, Architecture and Components
- System Communication
- RSA Authentication Manager Licensing
- Authentication Options
- Administration Topics
  - Structure; Policy
  - User Administration
  - System Administration
- Reports, Logs
- Self-service and Provisioning
- Software Authenticators
- End User Troubleshooting

RSA EMC²

3

Objectives

- To understand the fundamental RSA SecurID functions and processes
- To provide hands-on experience and explain RSA Authentication Manager Administration options and tasks
- To learn how to access the available reports and tools to assist in end-user support and troubleshooting
- To understand how to deploy and manage RSA SecurID authentication

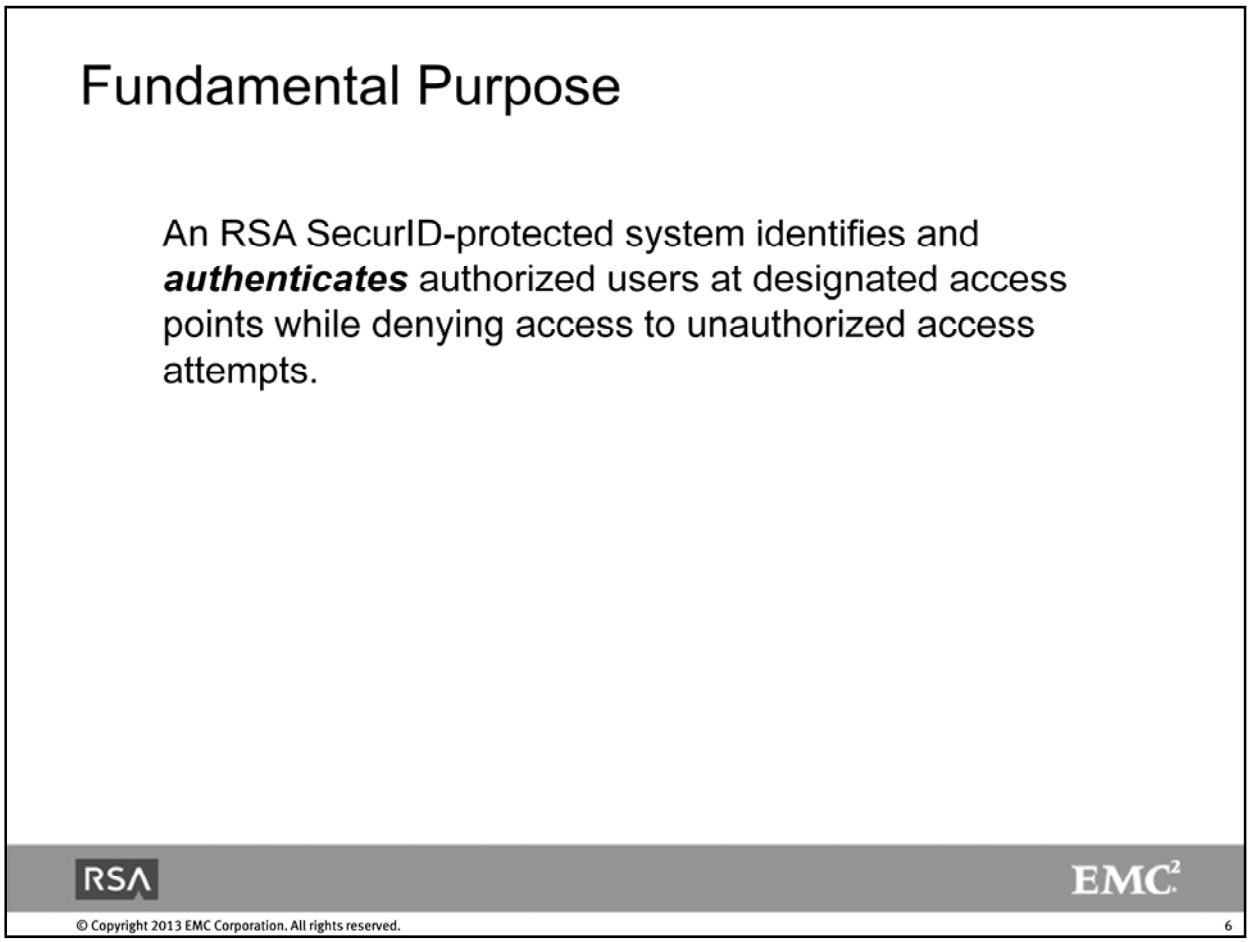
RSA EMC²

Product & Technology Overview

Unit 1

RSA EMC²

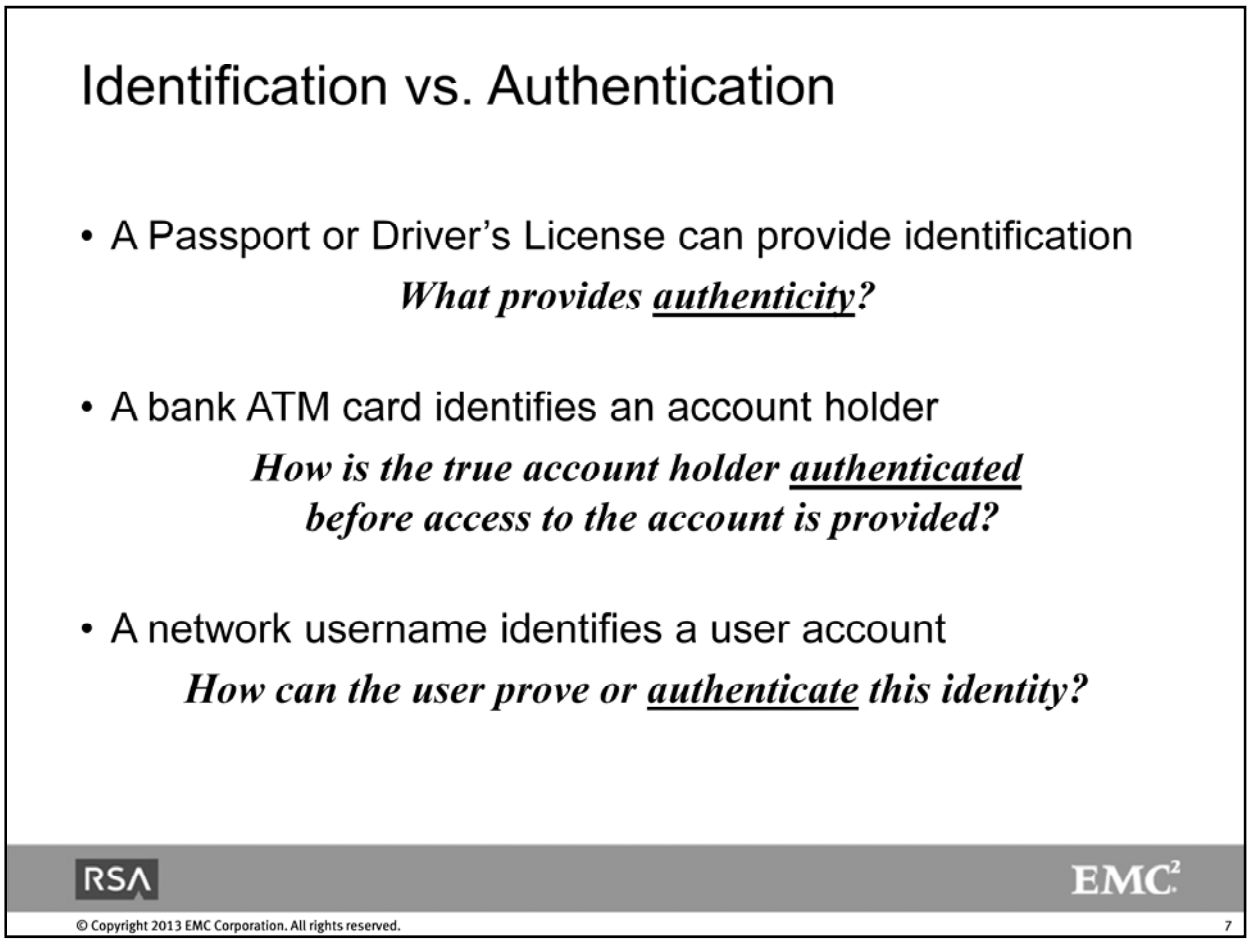
5



The purpose of an RSA SecurID system is to identify and authenticate authorized users on designated devices, while denying access to unauthorized attempts. Users are challenged for a unique Passcode through a cooperative network of RSA Authentication Manager servers and Agents.

The RSA SecurID Passcode provides a strong authentication mechanism to both determine access privileges and to provide positive, irrefutable identification of the user.

RSA SecurID supports access control by identifying and authenticating an individual which then allows that individual to be authorized for entry to a protected system or to access protected resources.



A passport or a driver's license is often used for purposes of identification. Certain mechanisms allow these documents to be used not only for identification but a picture or physical description and tamper-resistant construction typically support authenticity of the individual.

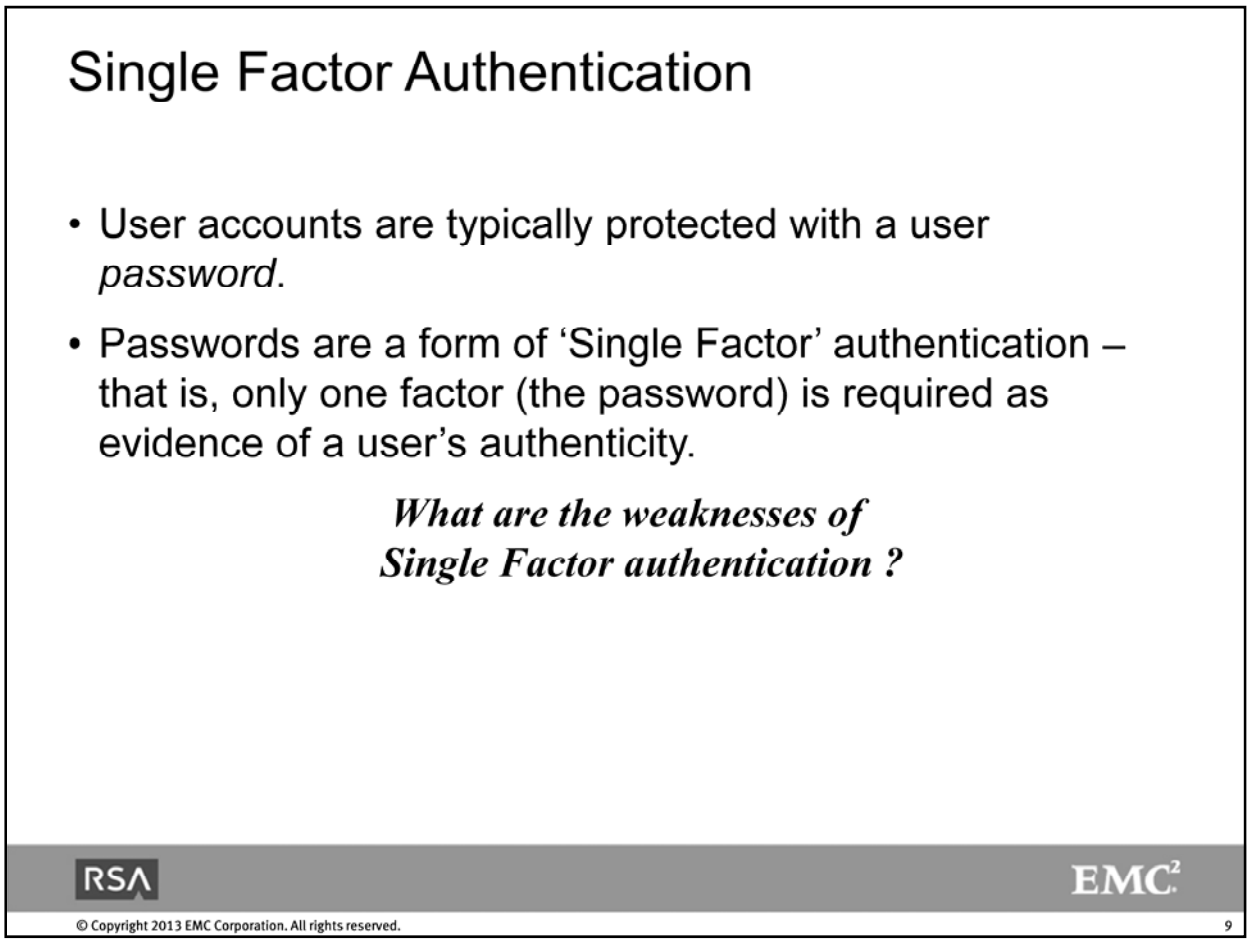
A bank ATM card requires the holder to provide a secret PIN as an authentication device before allowing access to account information or funds – the bank card alone provides identification that an individual is an account holder; the secret PIN in conjunction with the bank card offers authentication for the individual.

Exercise: Identification

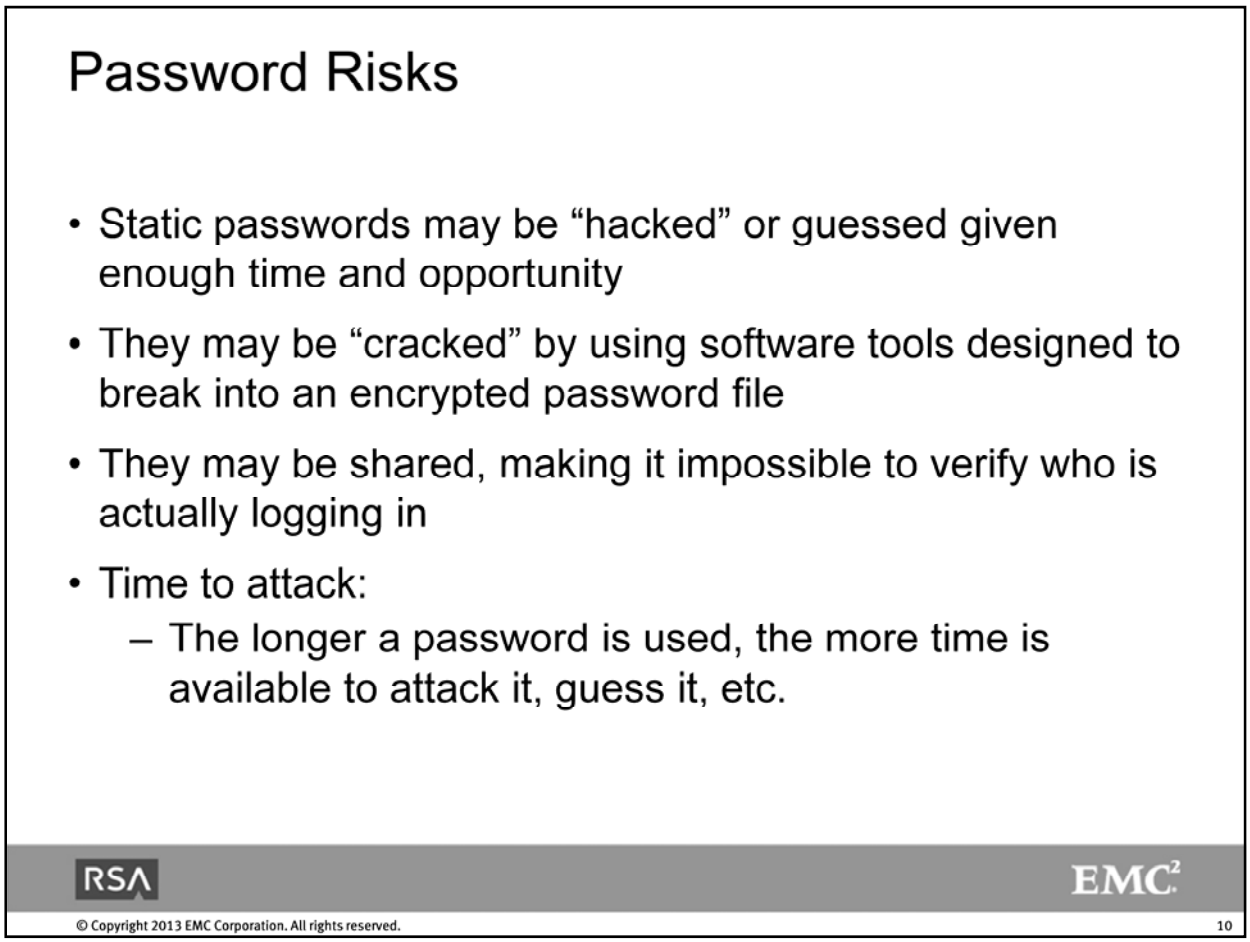
- Identify another member of your class through some simple means (business card, screen display - if in a virtual classroom, etc.)
- Discuss as a group what makes the various IDs trustworthy or not.
- What other supporting information would you need to trust the ID and why?

RSA EMC²

8



Network users often authenticate their login with a single authenticating factor - their network password. The use of a password as an authentication mechanism presumes that the password is secret - known only to the user who is presenting it to the system.



Risks to passwords stem from the fact that they may not always be “secret”. This occurs because passwords can be:

- “hacked” or guessed
- “cracked”
- shared among users (even innocently)

Static passwords

“Static” implies passwords are unchanged for substantial periods of time (days, weeks, months - sometimes indefinitely). This period allows a hacker time to try and guess a user's password. Hackers can employ social engineering methods to trick users into revealing their passwords or employ cracking tools.

Cracking passwords

“Cracking” involves the use of software tools to ascertain a user’s password. Such software can attempt to discover passwords by brute force (trying a series of numeric or alpha numeric combinations), monitor a computer for password keystrokes (“Trojan Horse” and “Key Logging” software), or various techniques to attack encrypted password files.

Sharing passwords

If passwords are shared, how can it be proven that a certain individual actually logged in to a given system at a given time? (For example: An executive leaves her Administrative Assistant her password to check e-mail during a vacation.)

A shared password may also not retain the same security priorities of the original user. (For example, the Administrative Assistant writes the password down and keeps it in an insecure location.)

Exercise: Bike Lock Attack

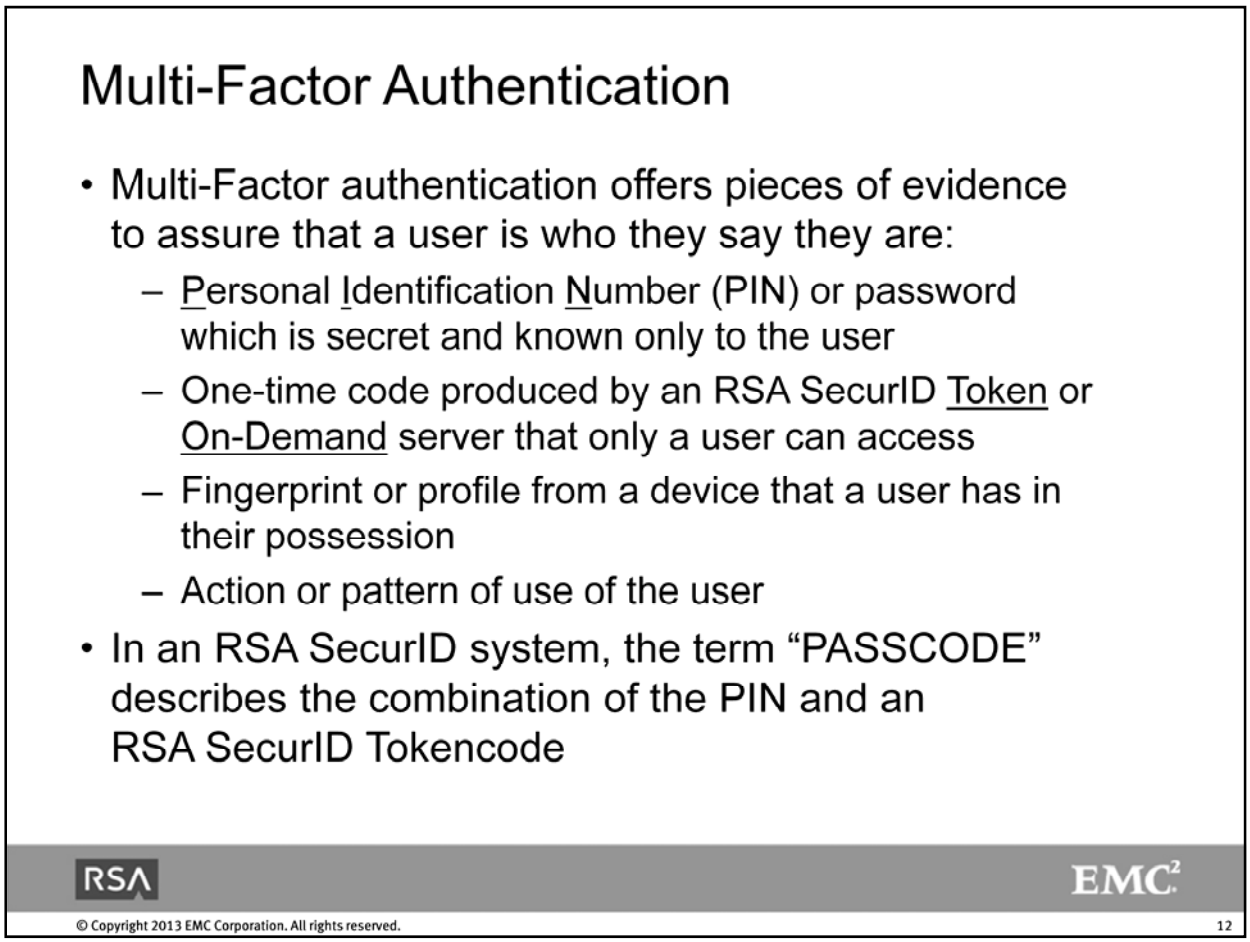
(Attaque de serrure de vélo)

- Assuming you can try 1 combination per second, How long will it take to brute force attack the 4-digit combination of a bicycle lock?
- What if the lock combination changed every 2 hours?
  Every hour?
  Every minute?

RSA EMC²

11

Everything sounds better en francais!



It is generally accepted that at least two factors are required to establish a "strong" authentication.

RSA SecurID authenticators provide two substantial factors:

Something you have or possess – the RSA SecurID authenticator or "Token".

Something that you secretly know – your Personal Identification Number (PIN)

An authenticator's code has no value without knowledge of the PIN; the PIN has no value without the authenticator. If each is reasonably guarded, then a user's authentication is extremely difficult to steal and infeasible to guess.

The fact that one and only one combination of PIN and authenticator can exist at any one time is a powerful tool to provide a way for individuals to authenticate themselves in a digital environment.

RSA SecurID Token Solution

- Something you have:
  - RSA SecurID Token/Software Token
    - Selectively issued to network users
    - Keyed to a specific user identity (user ID)
  - If stolen, account is protected by…
- Something you know:
  - Secret PIN
    - Only legitimate token holder should know the PIN
  - If PIN is compromised, account is protected by…
- Something you have:
  - Attacker must have *both* factors to mount a viable attack

RSA EMC²

13

The RSA SecurID system is based on two-factor "strong" authentication. Each user must provide the two pieces of evidence to prove that they are who they claim to be: something known, something possessed.

The combination of these two factors: the PIN, followed by the authenticator's displayed tokencode constitute an RSA SecurID "Passcode".

RSA SecurID On-Demand Code

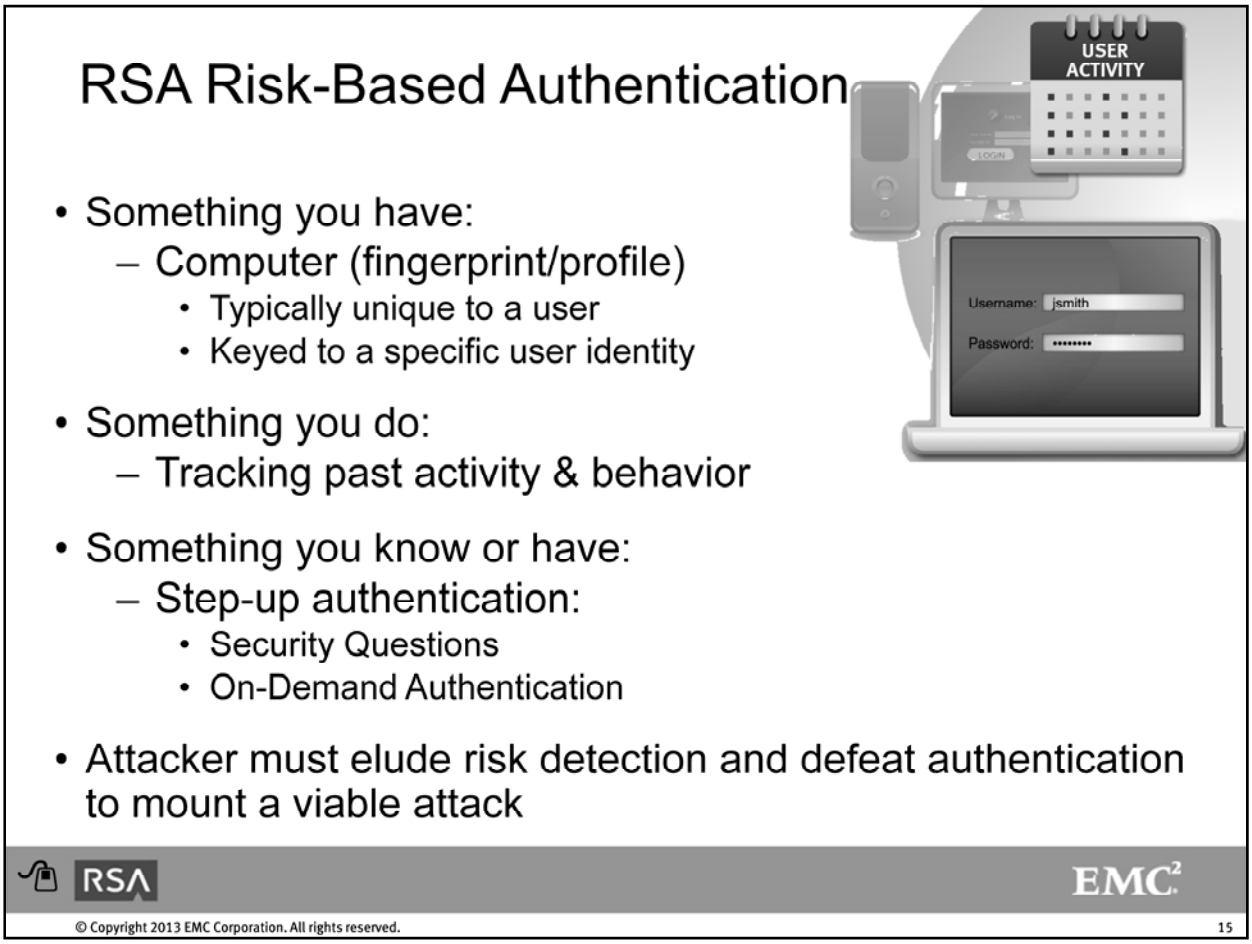
- Something you have:
  - On-Demand code on your phone or email
    - Selectively issued to registered users
    - Keyed to a specific user identity (user ID)
  - If stolen, account is protected by…
- Something you know:
  - Secret PIN
    - Only legitimate user's PIN can initiate On-Demand code
  - If PIN is compromised, account is protected by…
- Something you have:
  - Attacker must have *both* factors to mount a viable attack

RSA EMC²

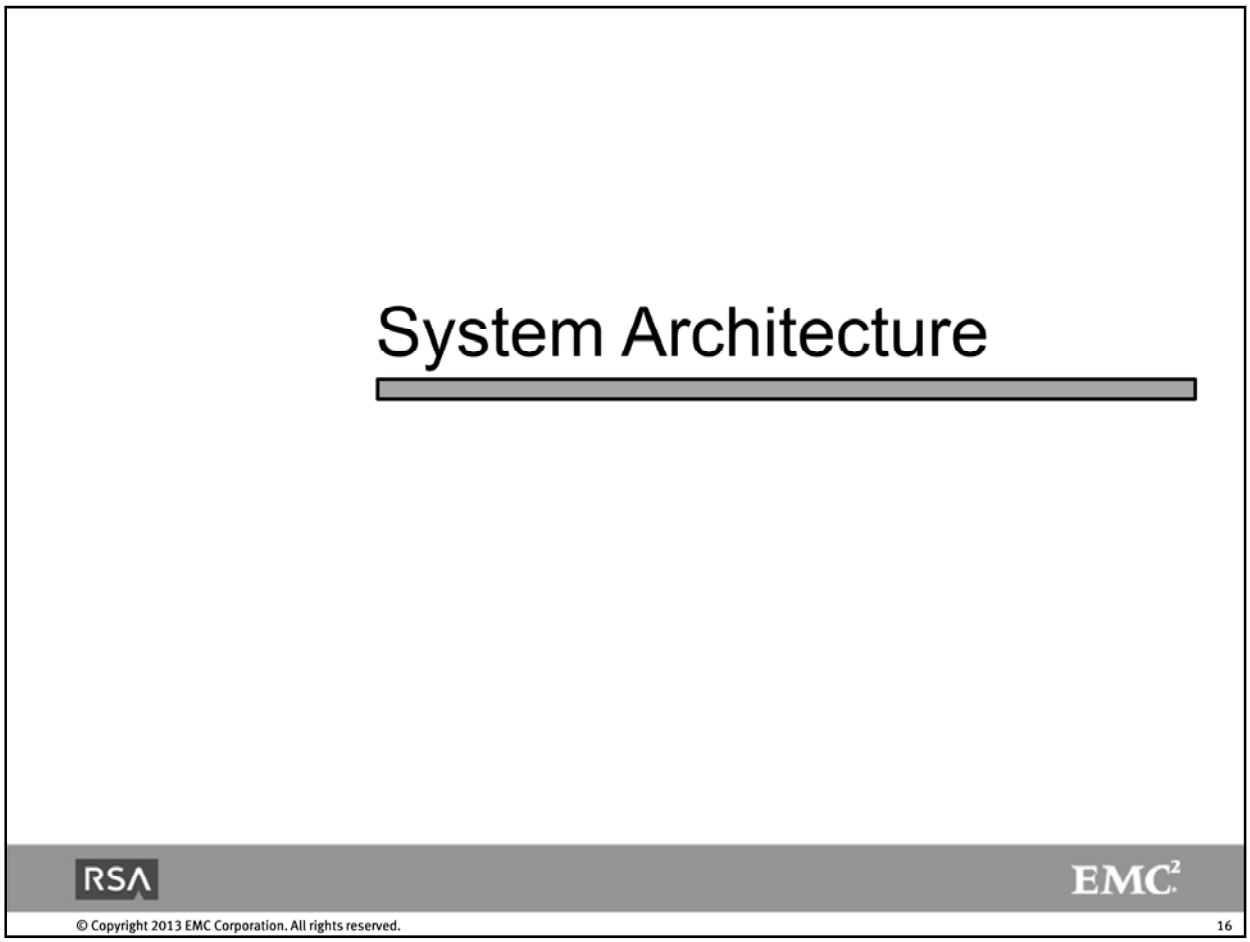
14

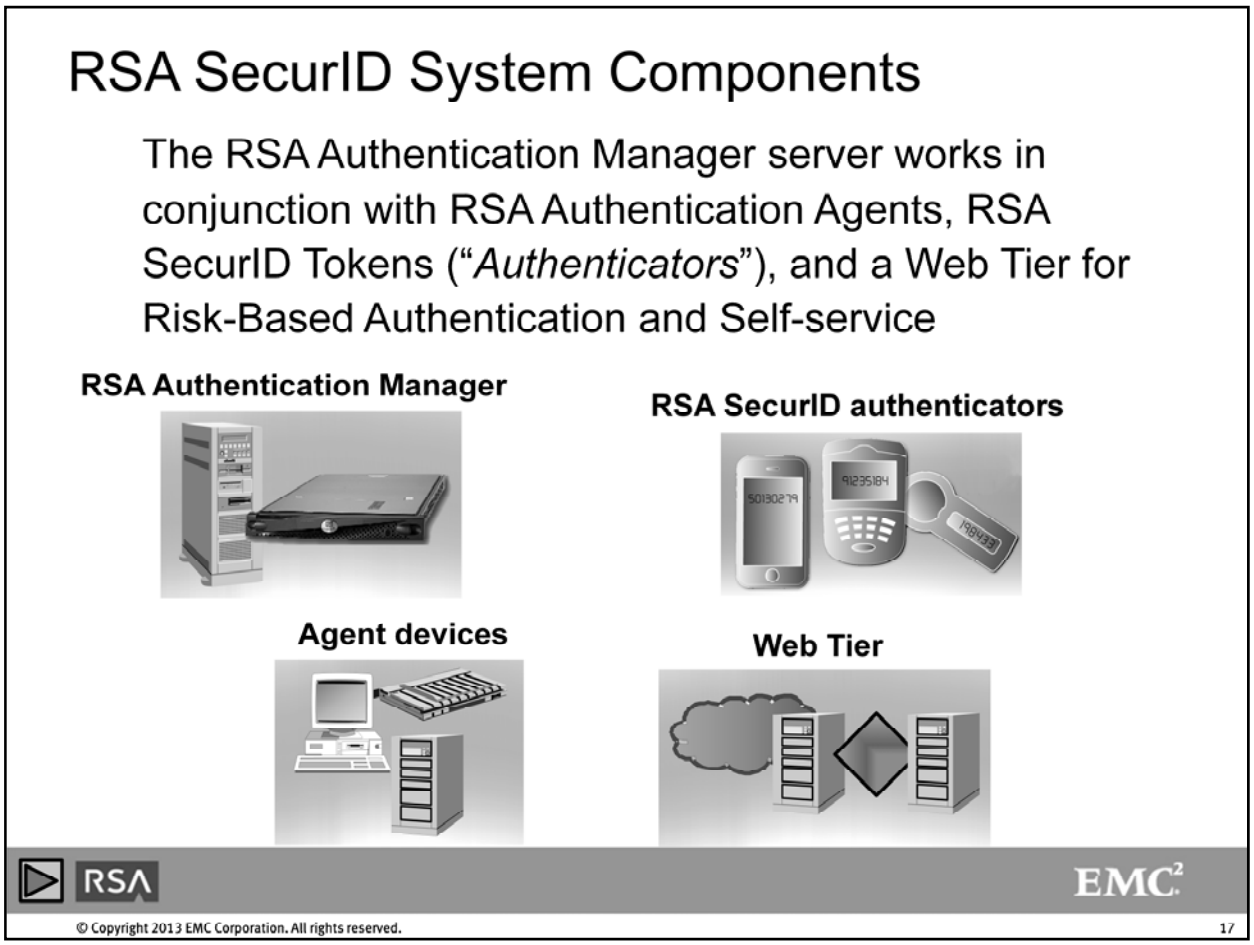
An On-Demand Authentication solution combines a PIN (to initiate an electronic transmission of a one-time code) along with the transmitted code itself to provide multiple authentication factors.

The possession of a registered device (mobile phone to receive the transmitted code) or registered e-mail account (to receive the transmitted code via e-mail) is another factor of On-Demand Authentication.



Risk-Based Authentication combines multiple factors by comparing a known device profile to what the user is using at the time of login, the user's behavior to see if it matches a past pattern, and adds additional authentication in the form of Security Questions or On-Demand Authentication if a risk threshold is exceeded.


System Architecture
RSA
EMC²
© Copyright 2013 EMC Corporation. All rights reserved.
16

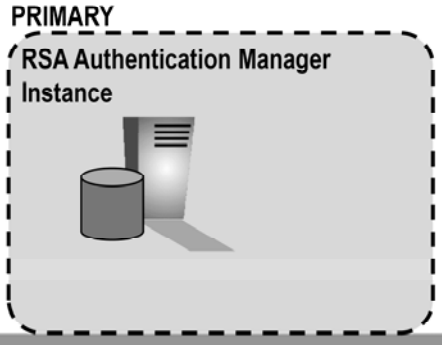


An RSA SecurID system has four primary components:

- RSA Authentication Manager
- RSA Authentication Agents
- RSA SecurID Authenticators or authentication methods
- RSA Web Tier

Authentication Server

- RSA Authentication Manager server is a virtual appliance deployed on a VMWare platform – ESXi or vSphere

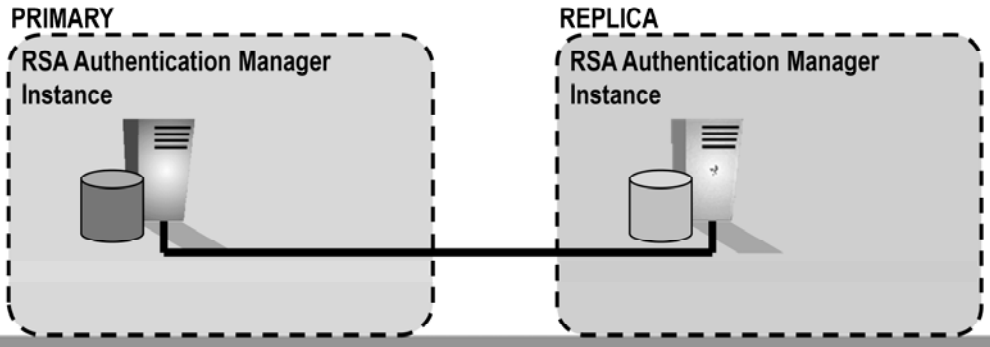


- First installation becomes the 'Primary' instance

RSA Authentication Manager is the server component of security client/server system designed to protect access to selected network resources. RSA Authentication Manager validates authentication requests from strategically placed Agents.

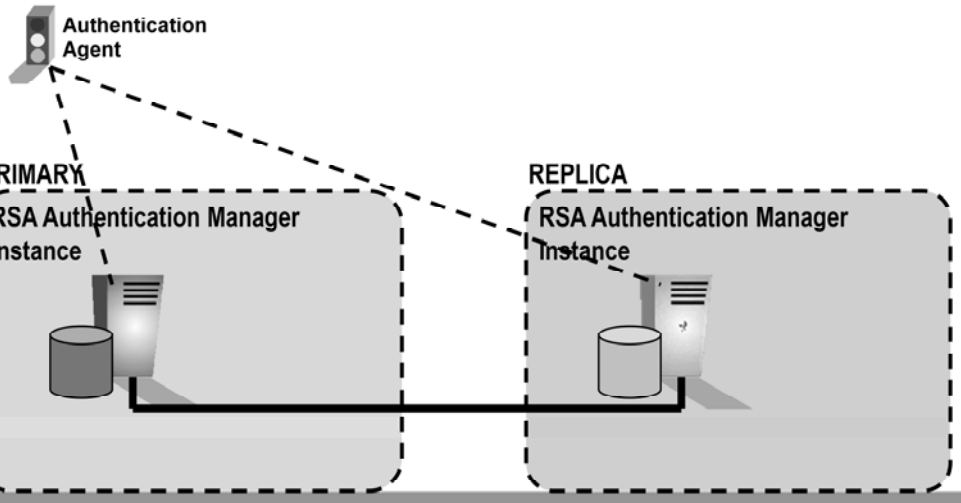
Authentication Server *(cont'd)*

- Up to 15 additional "Replica" instances can be attached for additional load balancing, fail-over and disaster recovery
- Replica servers handle authentication requests but do not allow administrative actions



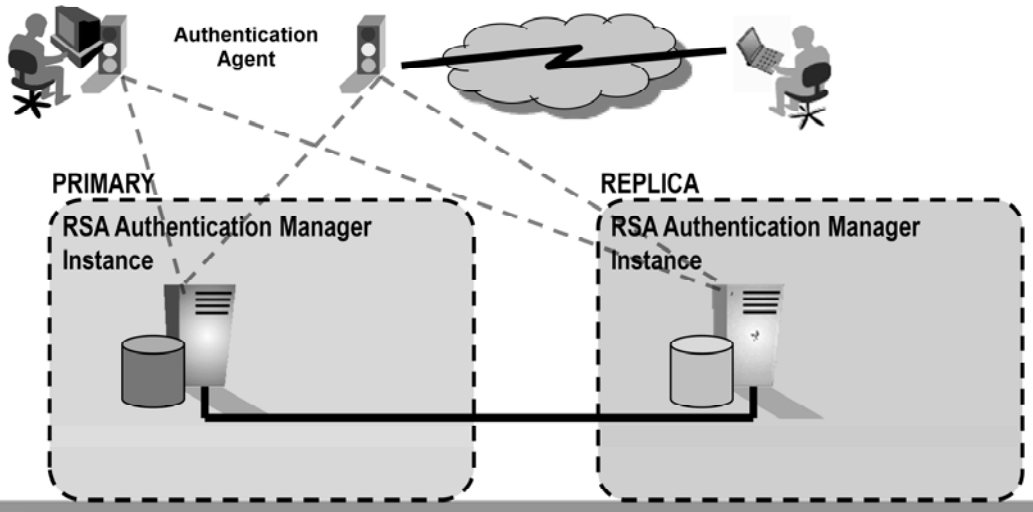
Authentication Server *(cont'd)*

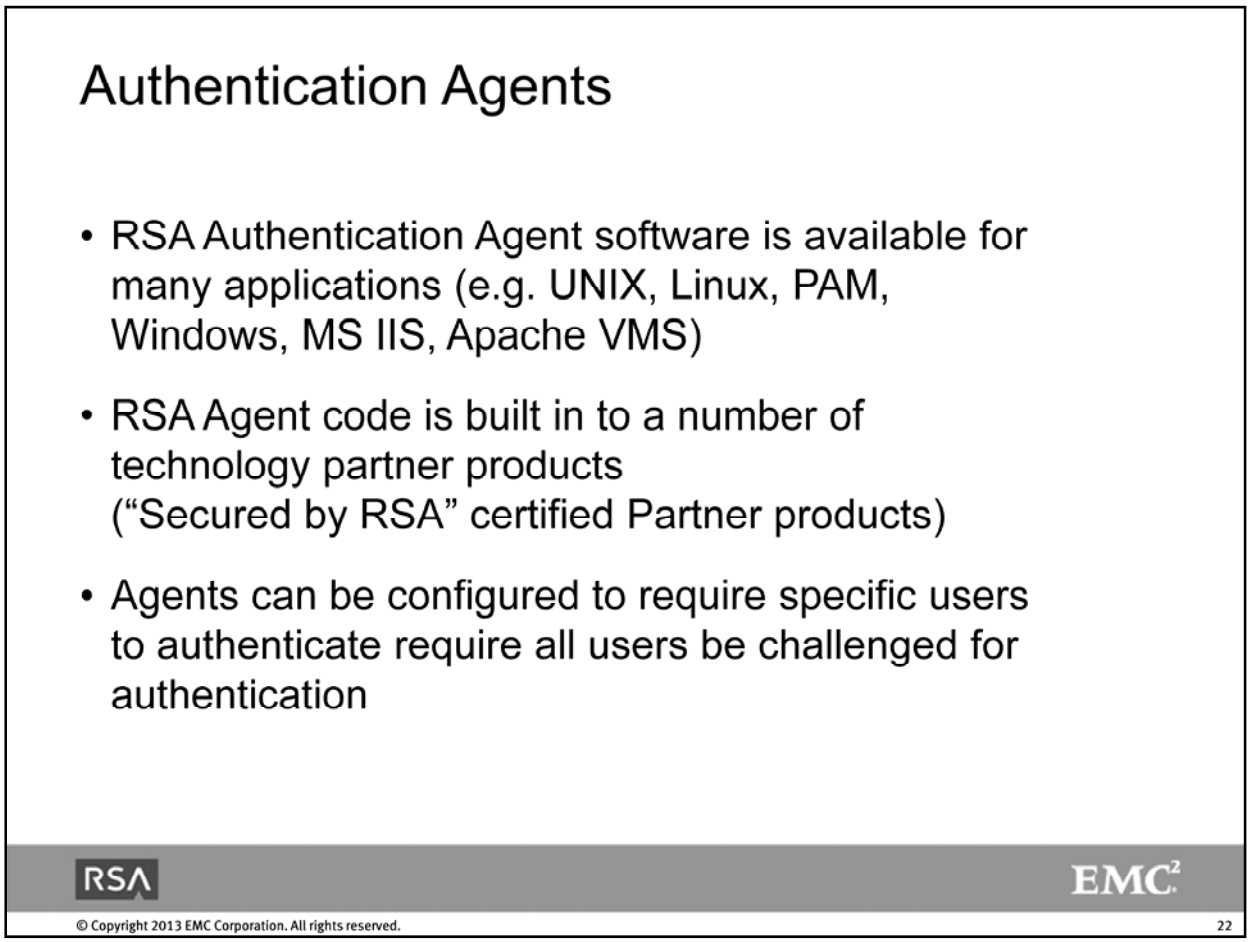
- Authentication Agents send requests to any, all or selected servers within the system



Authentication Server *(cont'd)*

- Users initiate authentication transactions either through a device with an installed Agent or via a network/internet connection that connects to an Agent

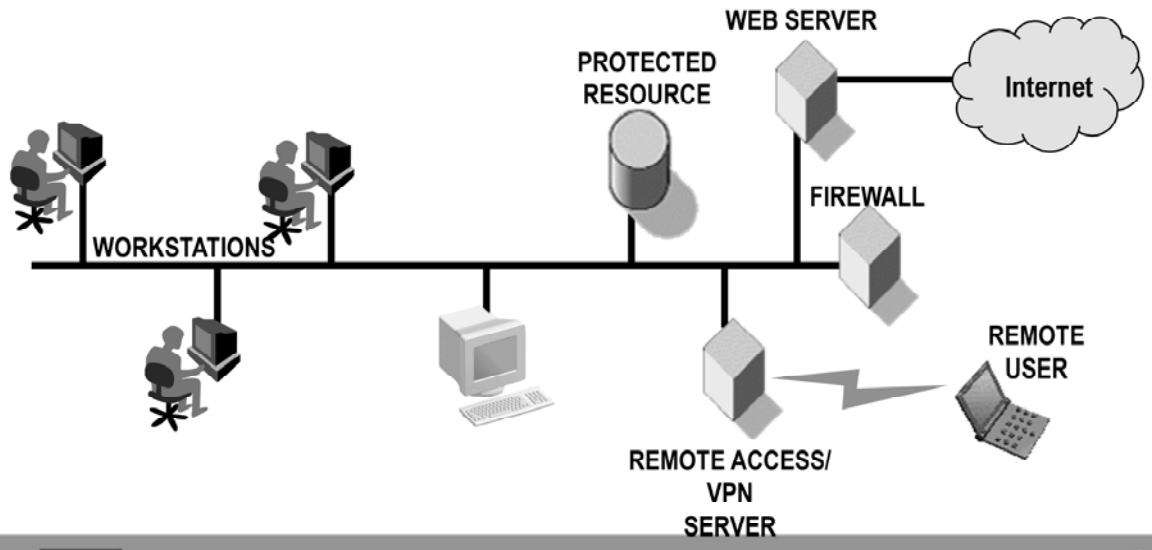




RSA Authentication Manager authenticates a user's identity based upon the user's supplied Passcode. It is the job of the RSA Authentication Agent to interrupt a user's login and obtain the user's Passcode – then communicate with the server to verify that Passcode.

System Deployment

Key resources and access points can be defined for a network…

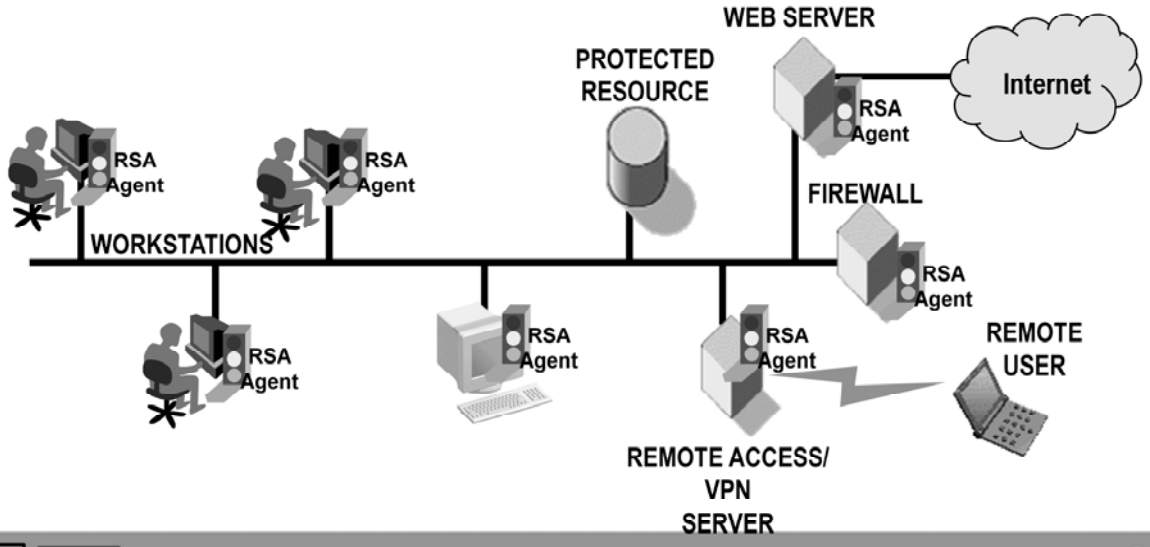


RSA EMC²

23

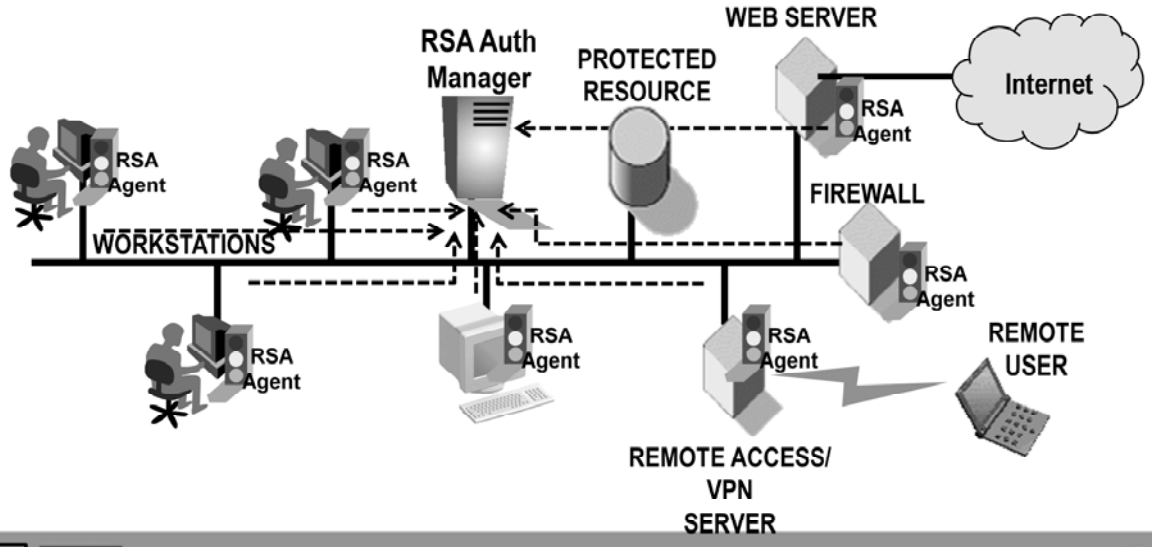
System Deployment *(cont'd)*

RSA Authentication Agents can be used at these points…



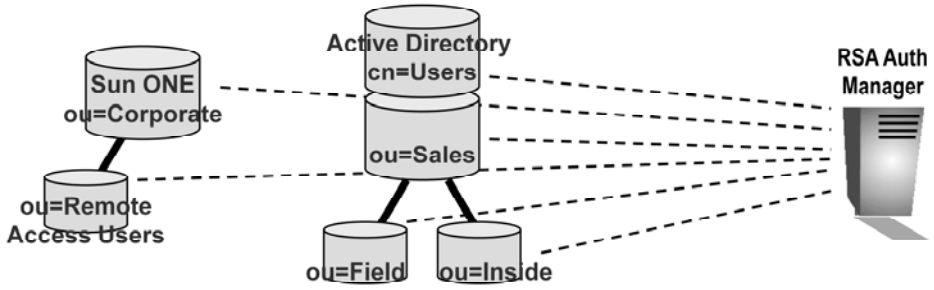
System Deployment *(cont'd)*

Agents send authentication requests to RSA Authentication Manager



System Deployment *(cont'd)*

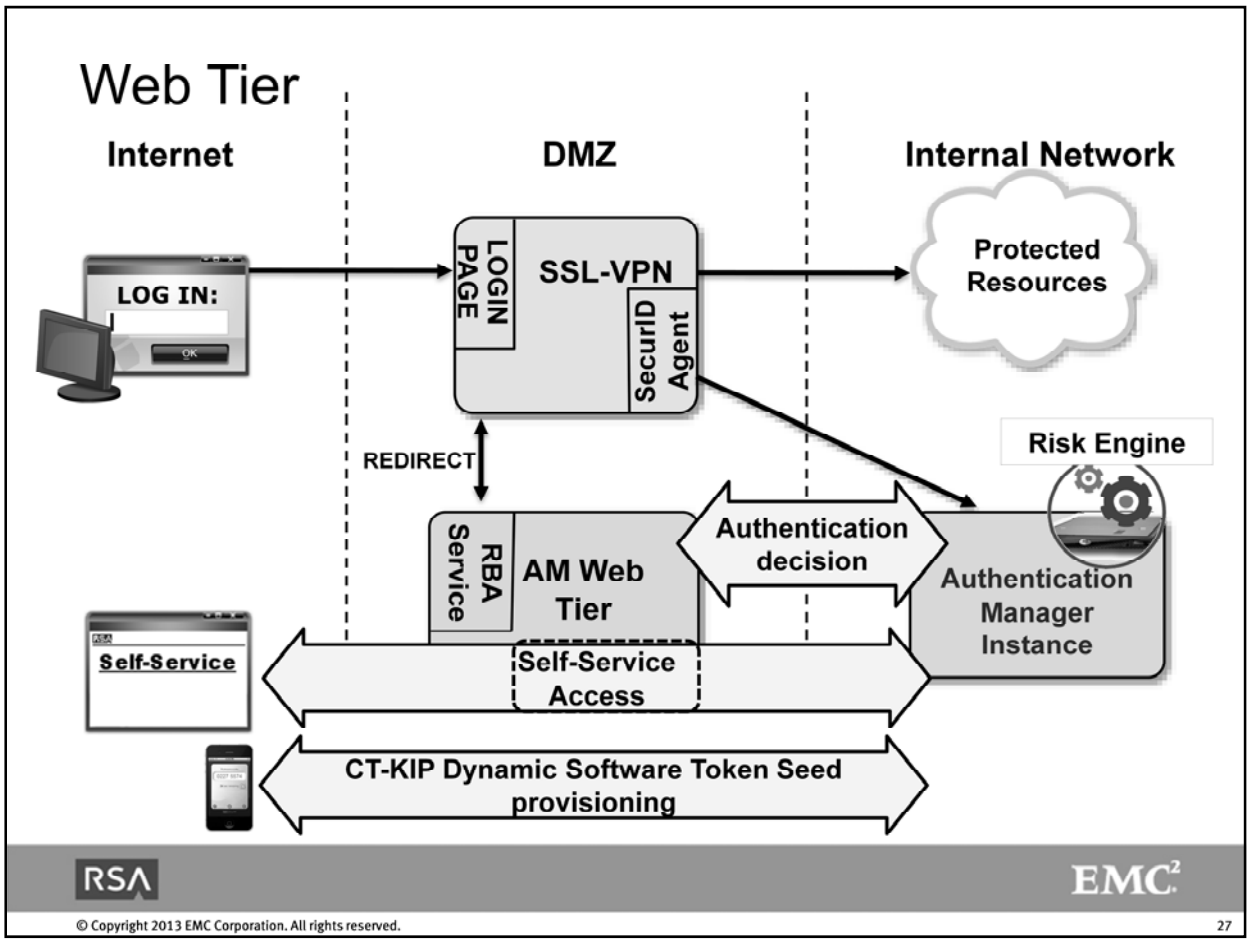
- A deployment can be associated with external "Identity Sources" such as LDAP directories for user accounts
  - Identity source can be the entire directory…
  - …or a part of the directory tree
  - Multiple Identity Sources can be configured



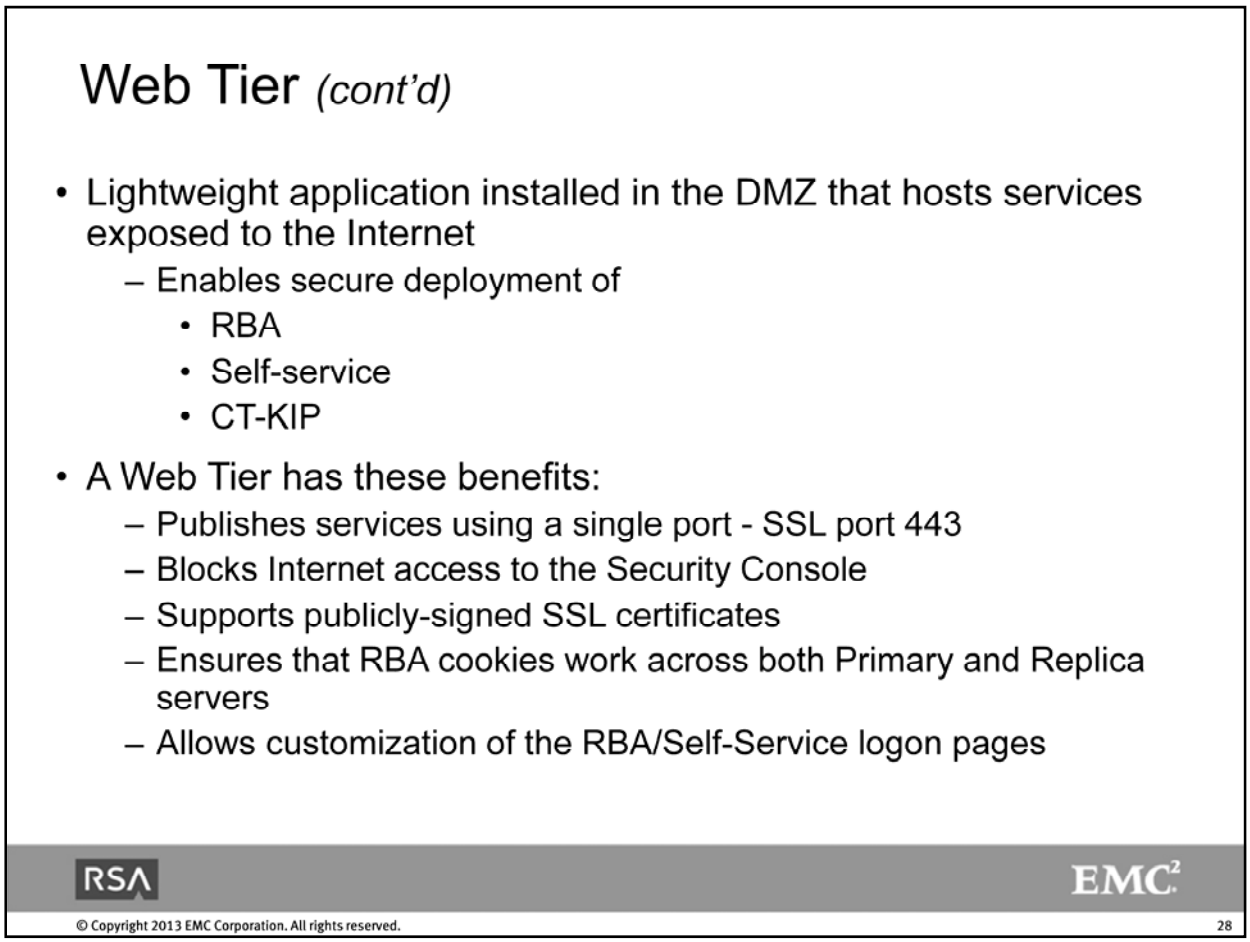
RSA EMC²

26

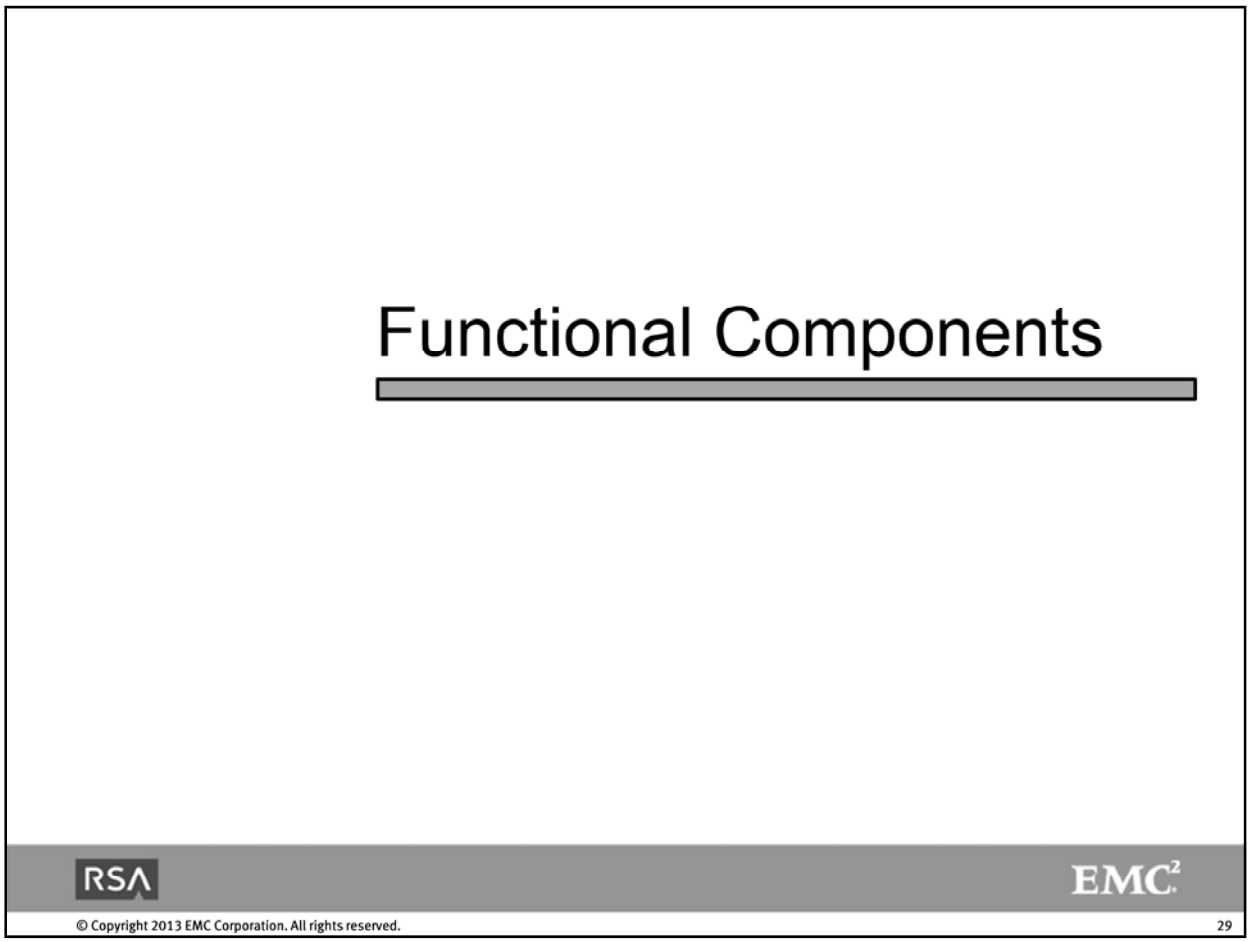
User accounts can reside in an internal database within the Authentication Manager virtual appliance, or user account information may be linked to existing information in an external directory server.



The RSA Authentication Manager Web Tier supports Risk-Based Authentication, access to the Self-service console, and CT-KIP Dynamic Seed provisioning for RSA SecurID software tokens. The major functions and communication paths are shown in this diagram.



The primary purpose of the Web Tier component is to isolate user connections from public space to an internal, private network. The Web Tier resides in the DMZ and allows external user connections – then communicates securely to Authentication Manager in the private network space.

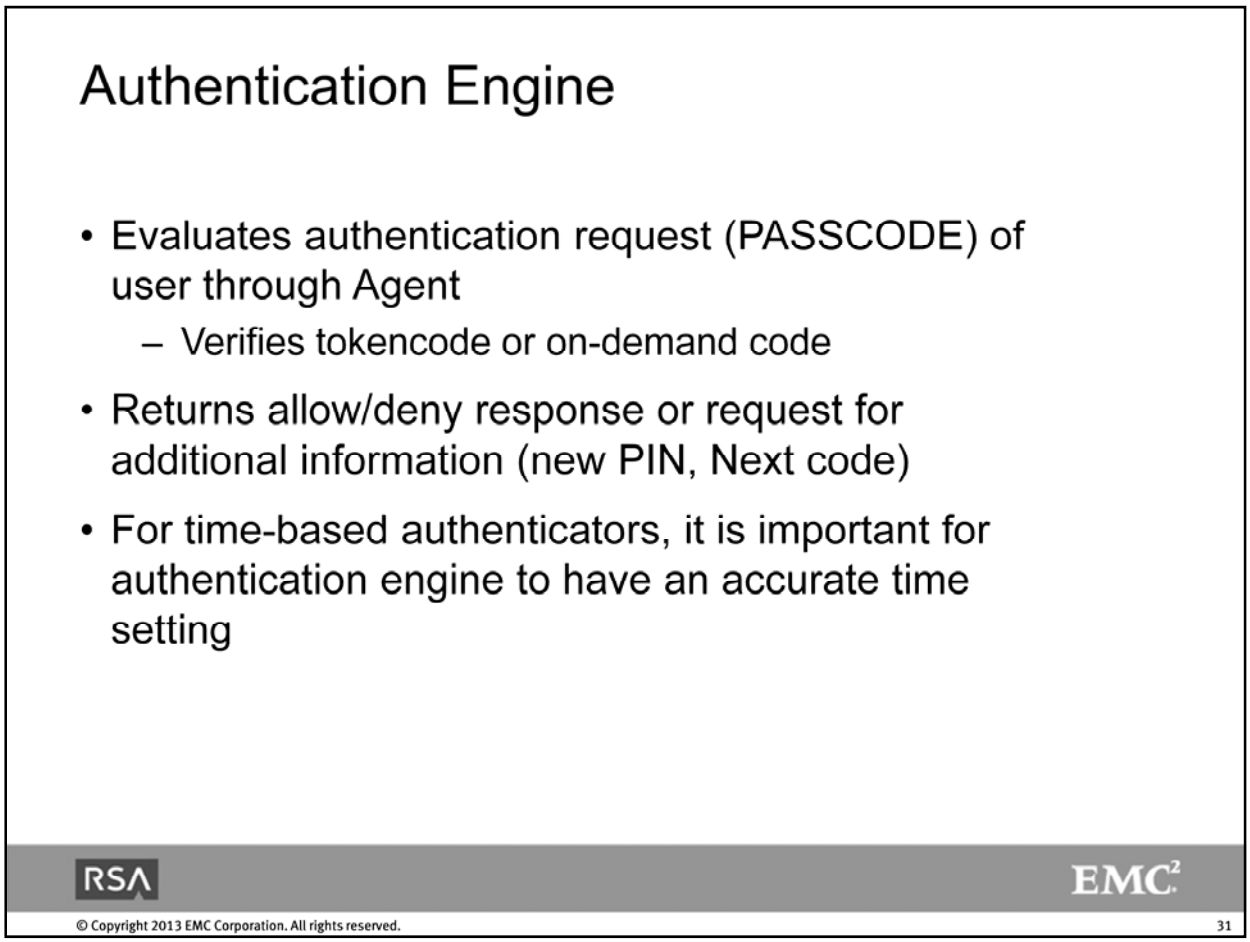

Functional Components
RSA
EMC²
© Copyright 2013 EMC Corporation. All rights reserved.
29

Functional Components

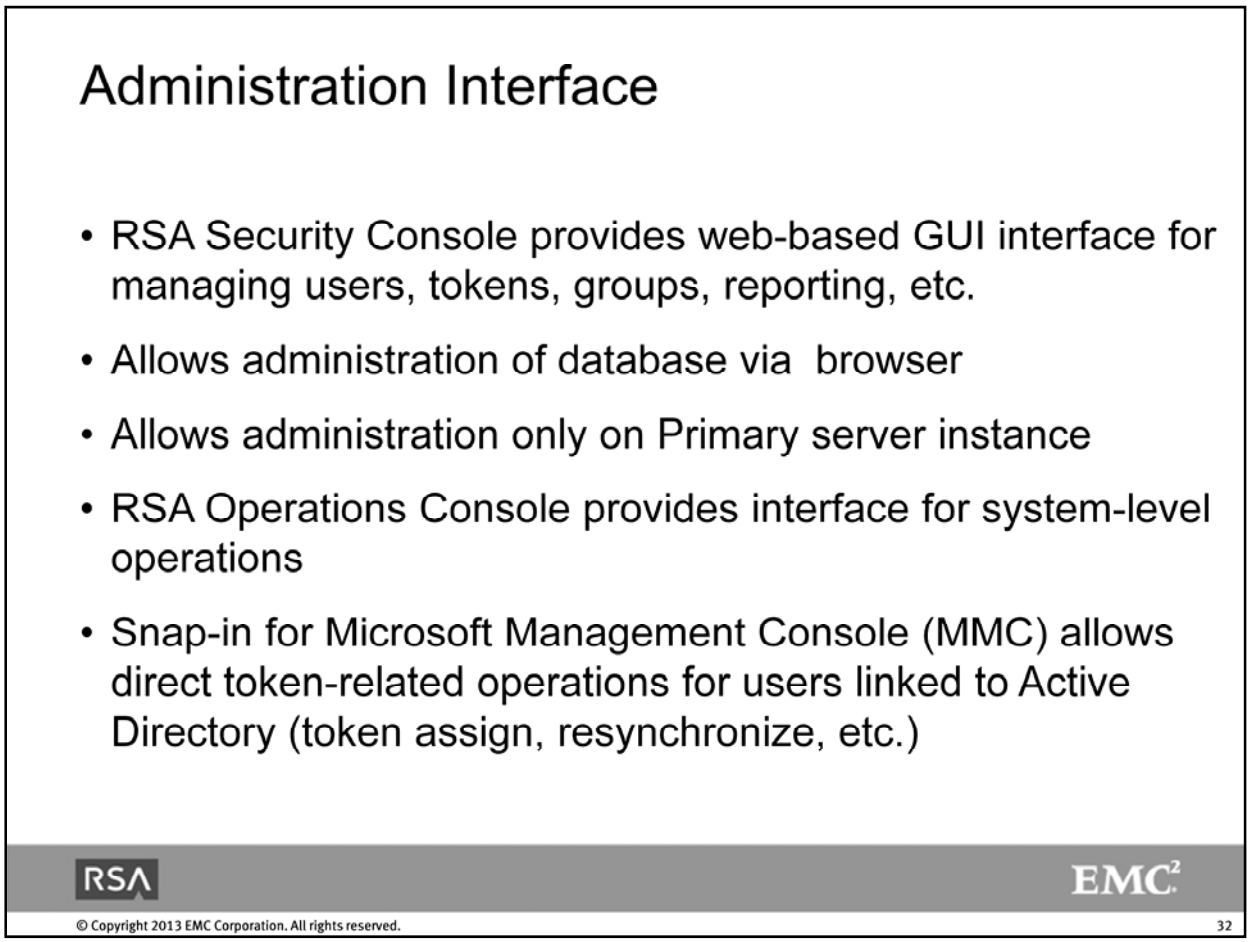
RSA Authentication Manager has four main functional components:

- Authentication engine 
- Administration console 
- Database 
- Risk Engine 

RSA EMC²

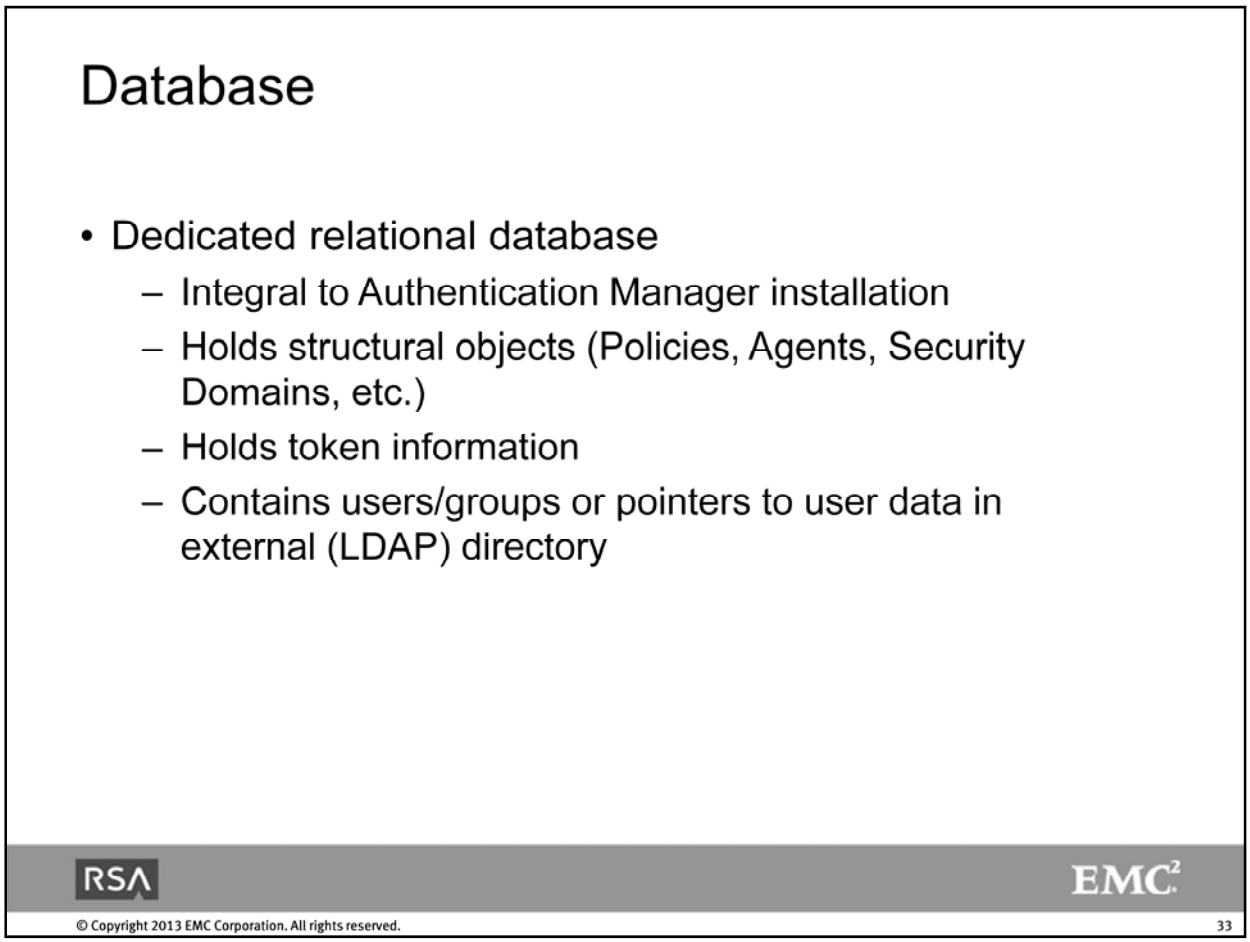


The authentication engine is the heart of the user authentication process. It works with Authentication Agent software to authenticate users as they enter the system.

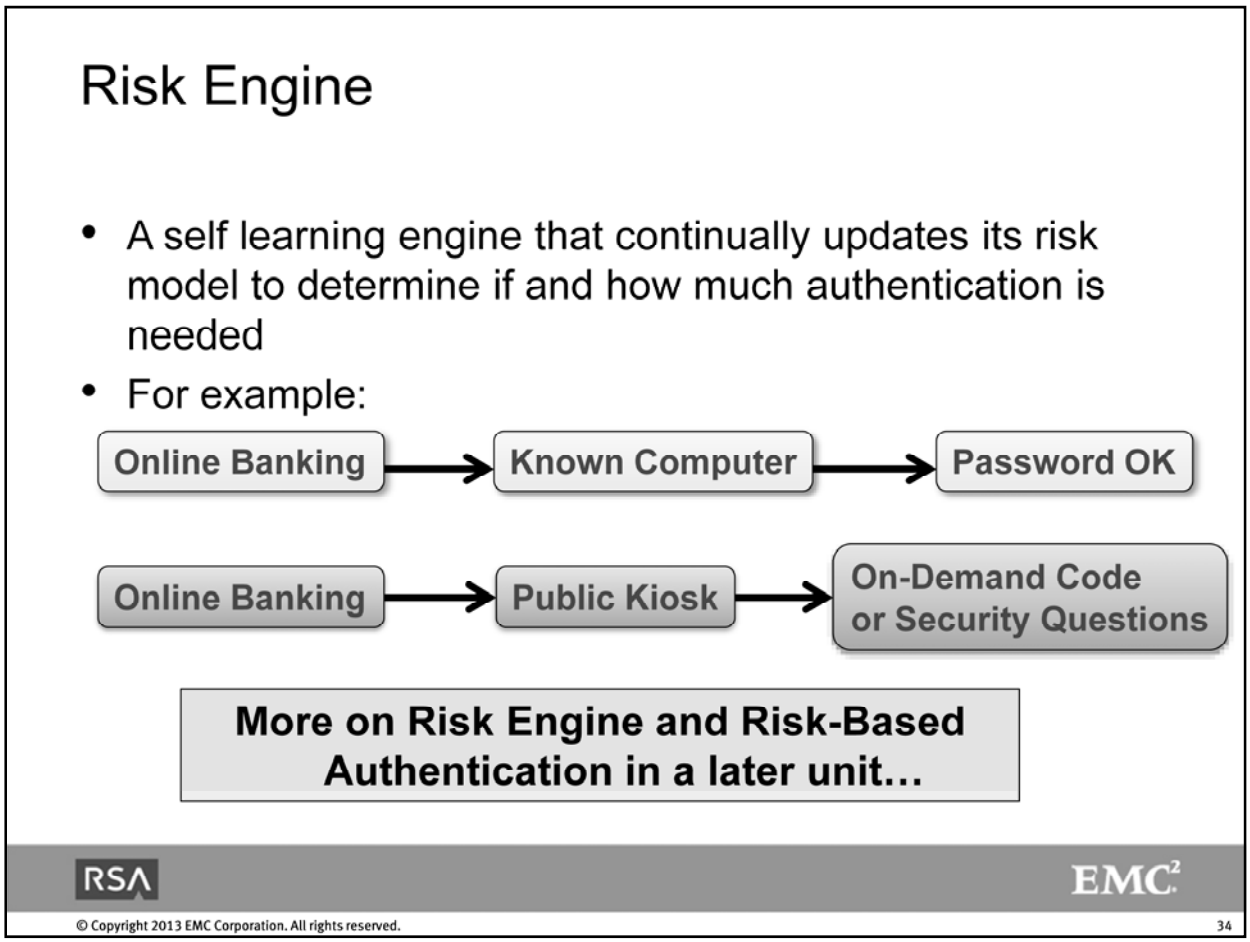


The RSA Security Console allows the system administrator to perform such tasks as adding users, modifying and assigning tokens, running reports and system maintenance tasks - all through a browser-based web interface.

Only authorized administrators are allowed to perform administrative tasks and various permission sets and scope of authority can be assigned to administrators.

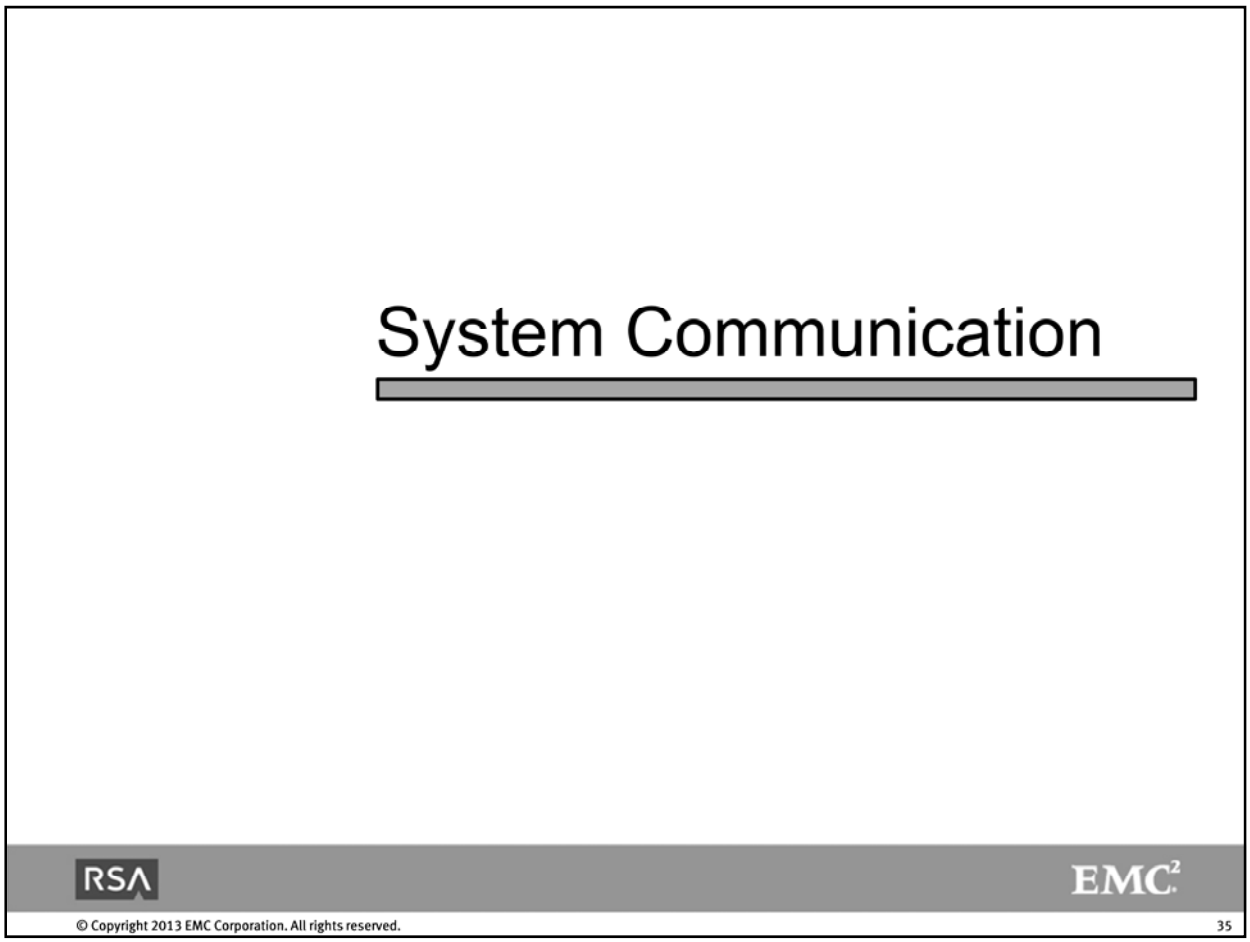


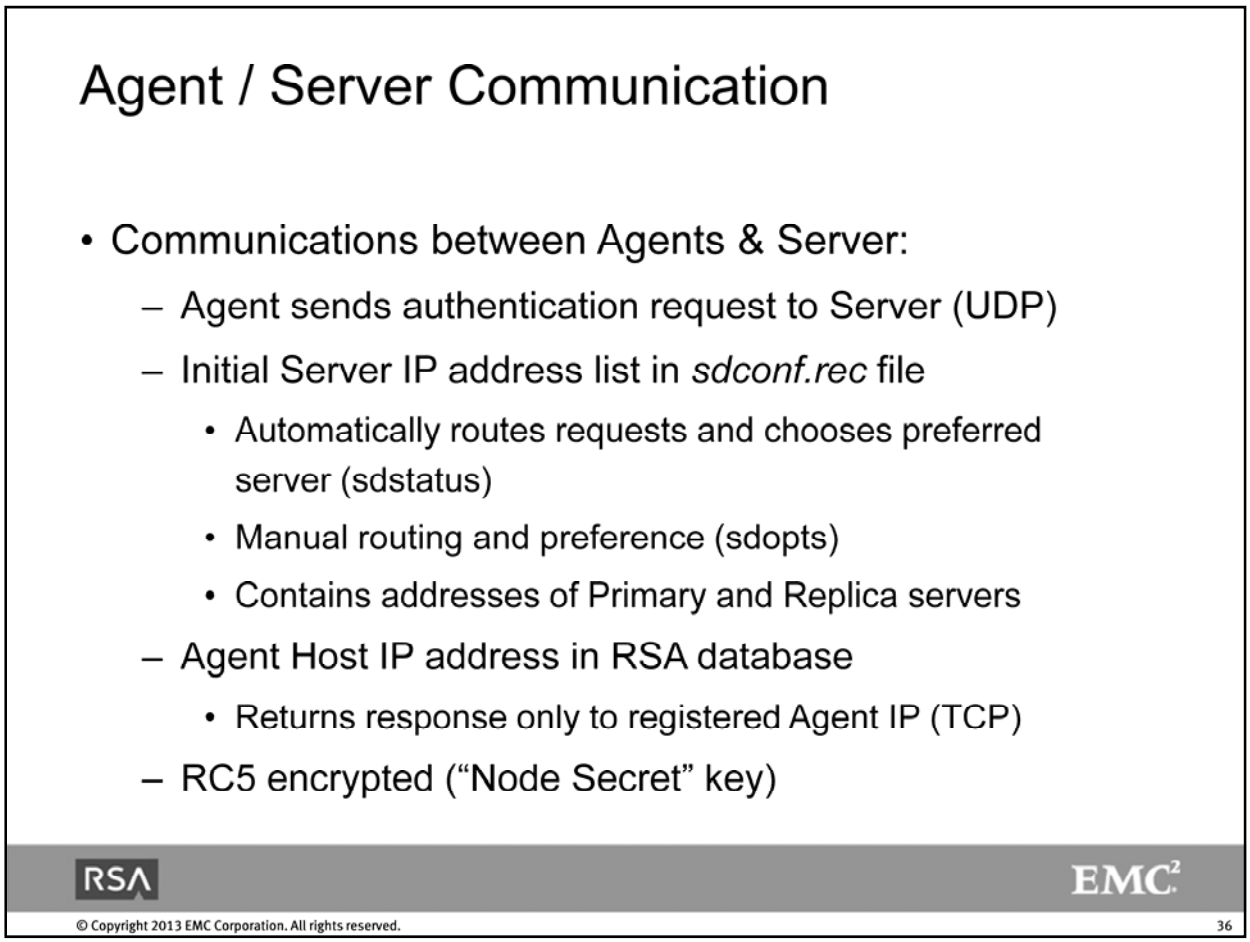
The RSA Authentication Manager Database is a relational database that holds user data and system entities (Tokens, Agents, Group structures, etc.) and audit logs that contain Authentication Manager events for user authentication and administrative activity.



The RSA Risk Engine examines a user's risk profile (past behavior, device profile, etc.) and compares it against an assurance level to determine if further authentication challenge is needed.

Users typically accomplish authentication using a username and password. If the risk engine determines that enough difference exists in past patterns, additional challenges are made (Security Questions or On-Demand Authentication).


System Communication
RSA
EMC²
© Copyright 2013 EMC Corporation. All rights reserved.
35

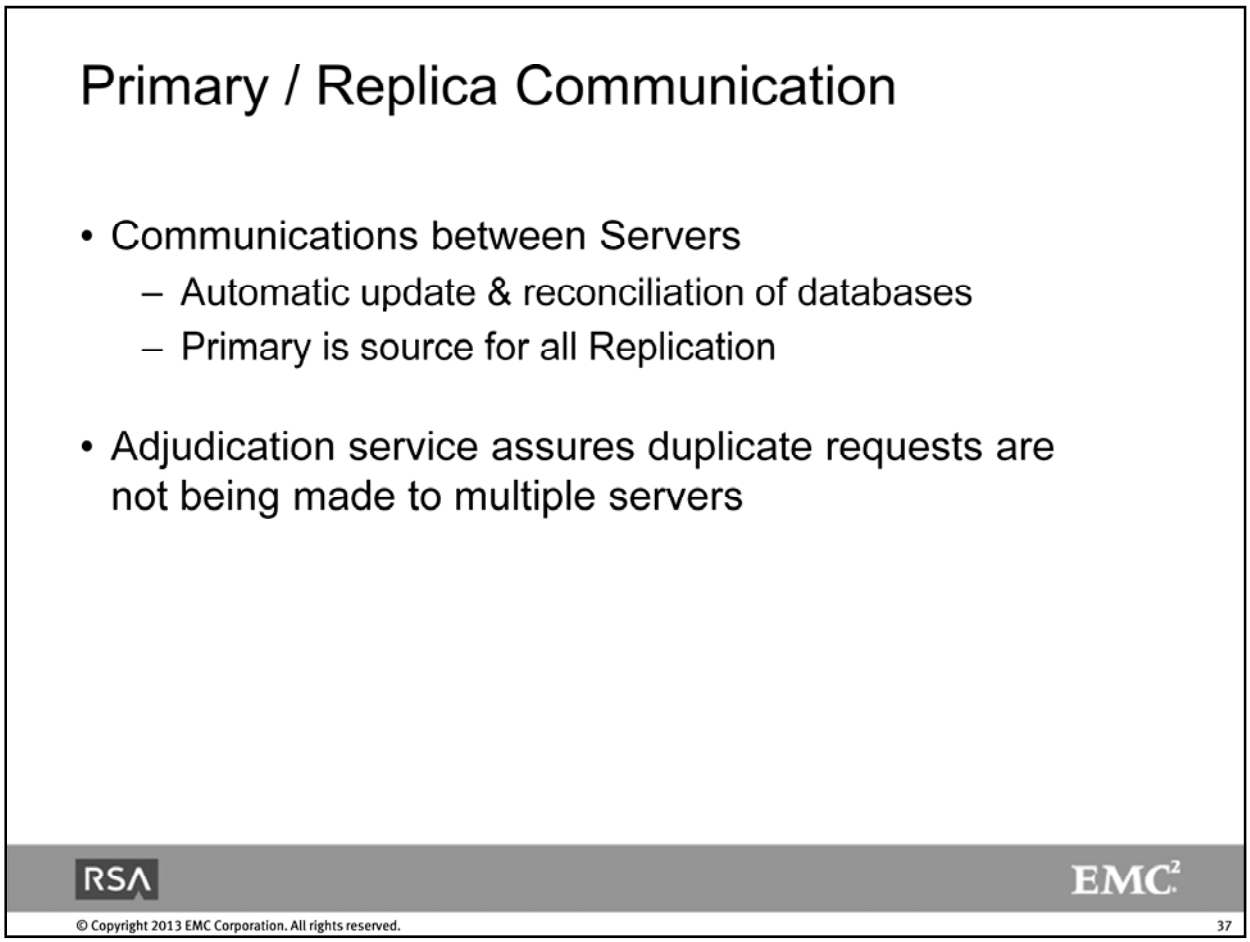


Agent/Server communications use UDP/Unicast for transmission.

Specific service ports are set up during installation to allow Agent-Server communication. By default, the Server authentication port has the name “securid” at port 5500 but may be renamed as desired as long as all such ports within the system use the same port identification.

The Primary Server address is contained in the Agent’s sdconf.rec file and will be the first Server contacted by a new Agent. In the initial contact, the Primary Server will supply the Agent with a list of all active Servers in the system. After learning all the Servers, the Agent will build an sdstatus file and determine response times from each Server. This action is automatic and is periodically tested and adjusted, if necessary, by the Agent. The sdstatus information can be manually overridden by the use of an sdopts.rec file. The sdopts.rec file contains Server contact information with the administrator's preference for priority.

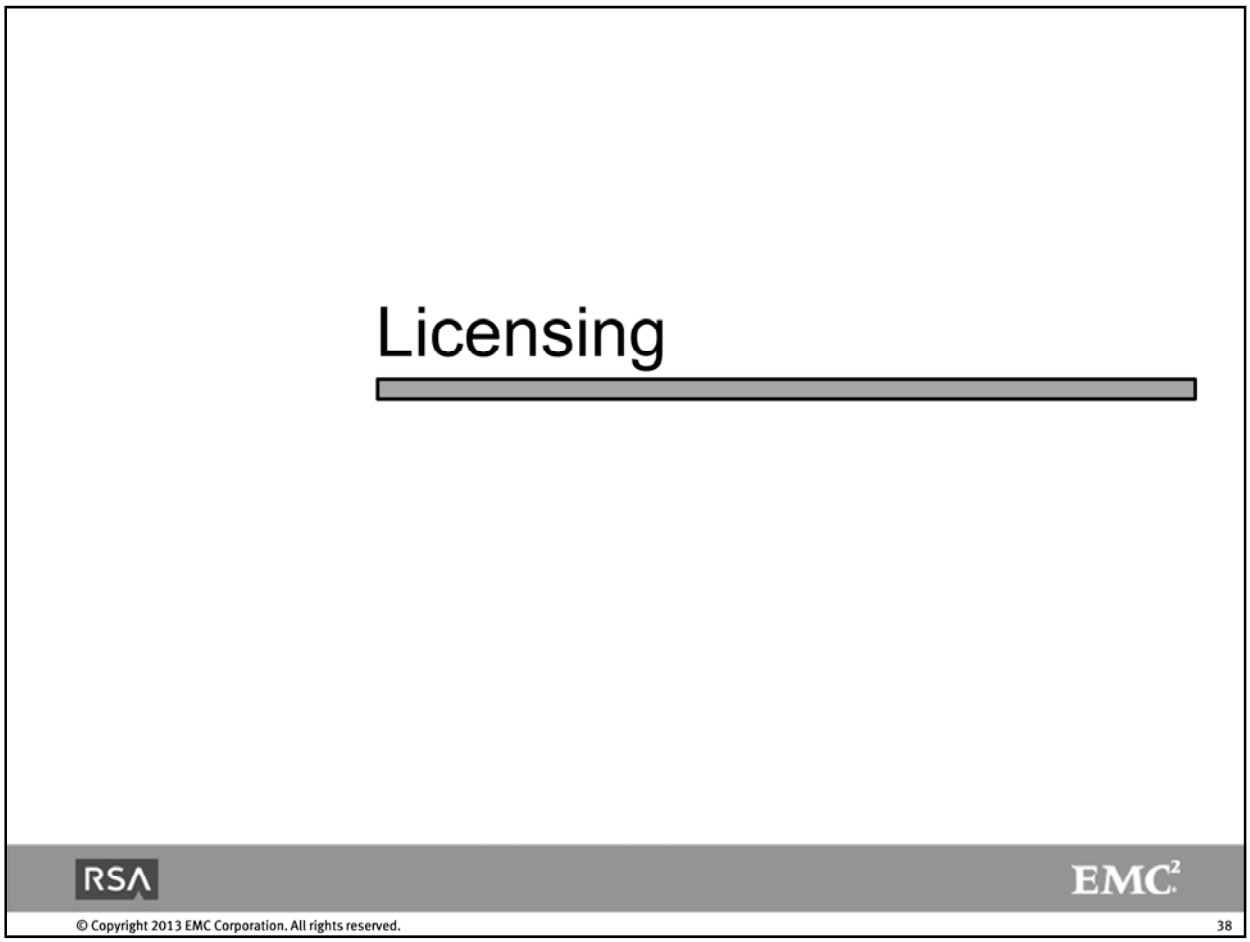
Agent/Server communications protect against 'masquerading' by utilizing the source/target IP addresses. The IP address with which to contact the Server(s) is part of the Agents sdconf.rec configuration file. The IP address of the Agent is held as part of the Agent's profile in the Server database. Servers will only accept Agent communication from the IP Address it “knows”. IP Address translation, such as through a firewall or router, might be problematic for such communication but configuration of a second (“Alternate IP”) address is allowed for an Agent so that secure communication can take place.

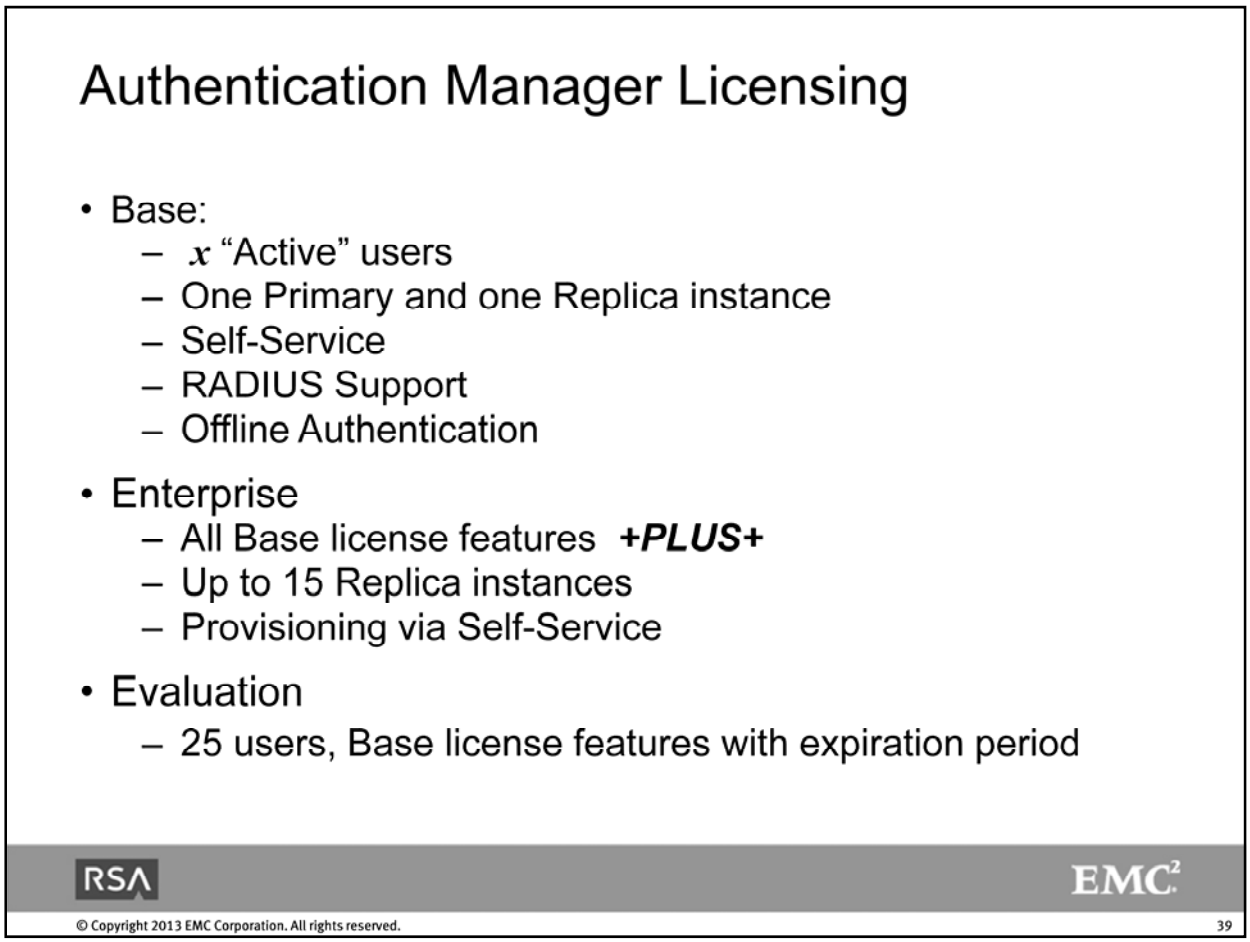


Primary/Replica Server communications use TCP for transmission.

Database information is shared among Servers by using the Primary Instance as a hub to process updated and changed data. Any operation at the Primary is propagated to the Replica databases; any change that occurs at a Replica Instance (New PIN, user disabled, etc) is updated at the Primary database.

The adjudication service operates between Primary and Replica Server Instances to prevent replay attacks.


Licensing
RSA
EMC²
© Copyright 2013 EMC Corporation. All rights reserved.
38



Three license types are available for RSA Authentication Manager:

- Base
- Enterprise
- Evaluation

Base License

This license level allows one Primary Instance and one Replica Instance. An active user limit is assigned and is specified at the time of purchase.

"Active" users are users who have assigned credentials - either an RSA SecurID authenticator or Fixed Passcode (the password assigned to the user account for Self-Service access does not make the user an 'Active user' for license purposes).

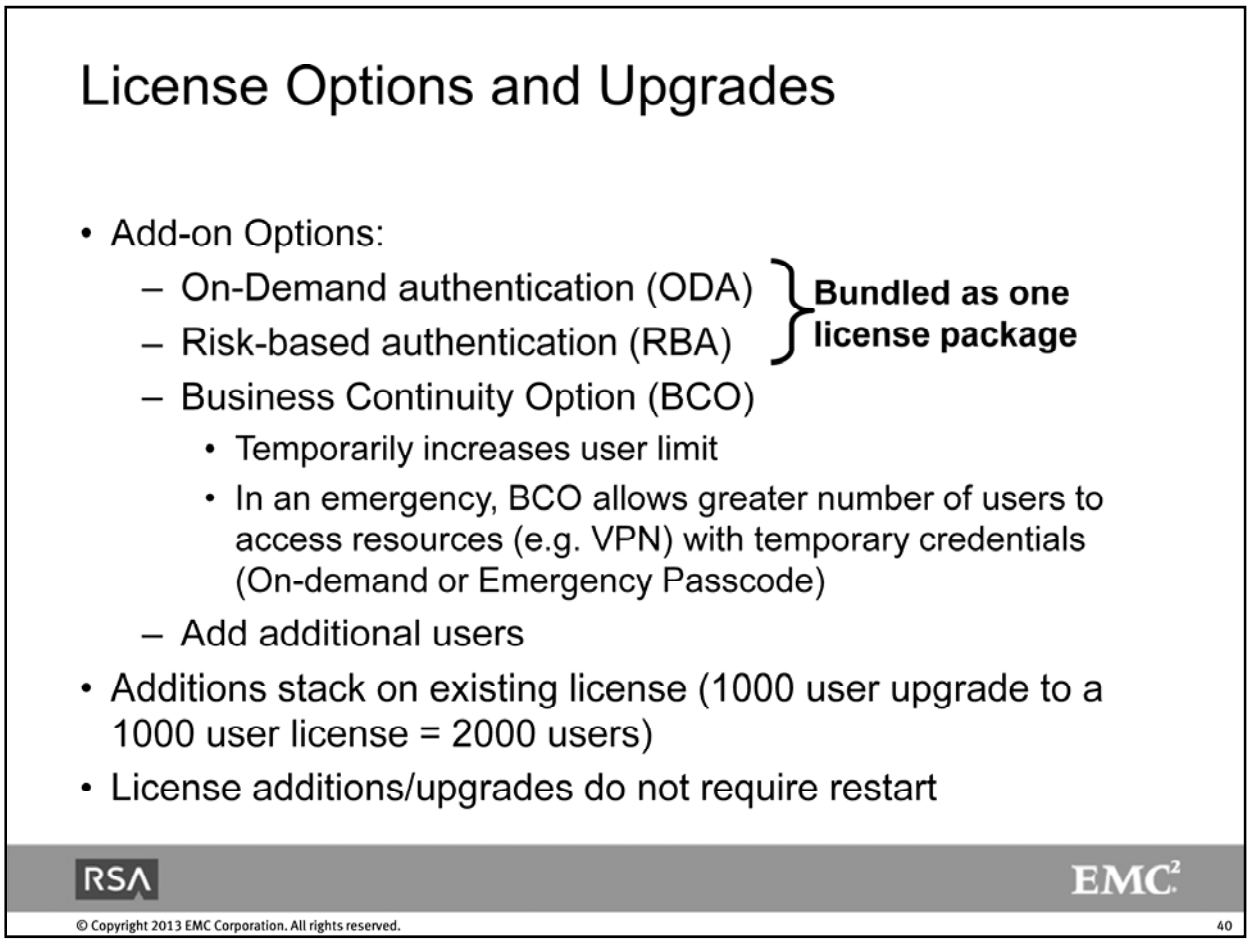
In addition. the Base license includes RADIUS support, support for Offline Authentication, and use of Self-Service (but without provisioning).

Enterprise License

The Enterprise license allows all of the functions of the Base license and allows the addition of up to 14 Replica instances to the deployment and provides full provisioning functionality for Self-Service.

Evaluation License

The evaluation license is designed to support a trial period for new and potential customers and allows a 25 user limit, all Base license functionality, and a defined expiration date.



Several options can be added to the initial Authentication Manager license:

Active User upgrades allow the system to be expanded for more users.

On-Demand Authentication allows the capability for a user to receive a one-time-use passcode through SMS [text] message to a device such as a cell phone or through e-mail.

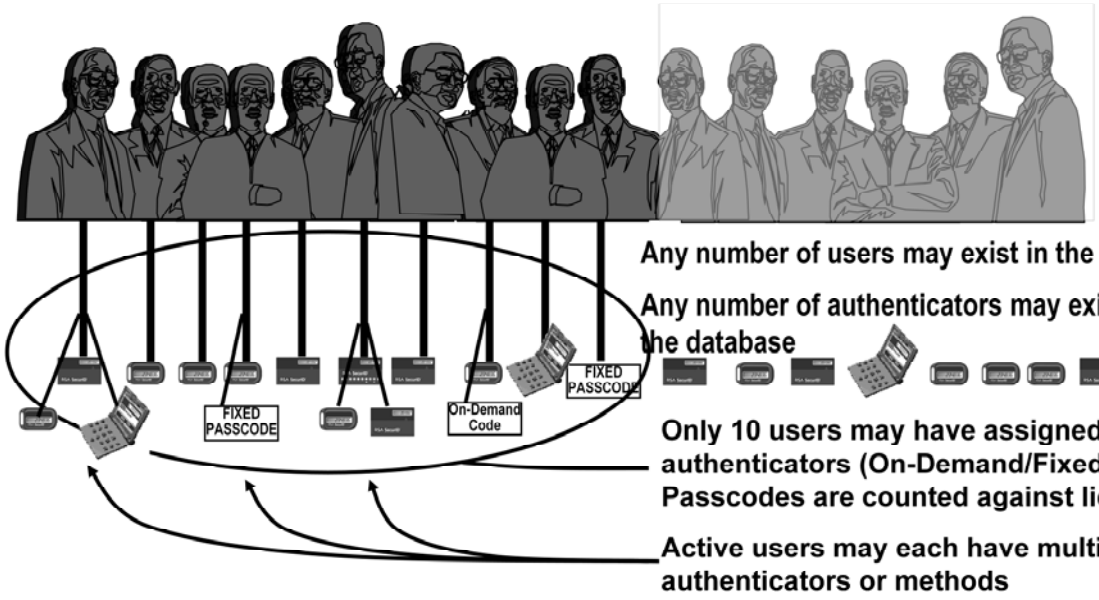
Risk-Based Authentication adds the Risk-based capability to enable user accounts and is bundled with On-Demand authentication as a single license option.

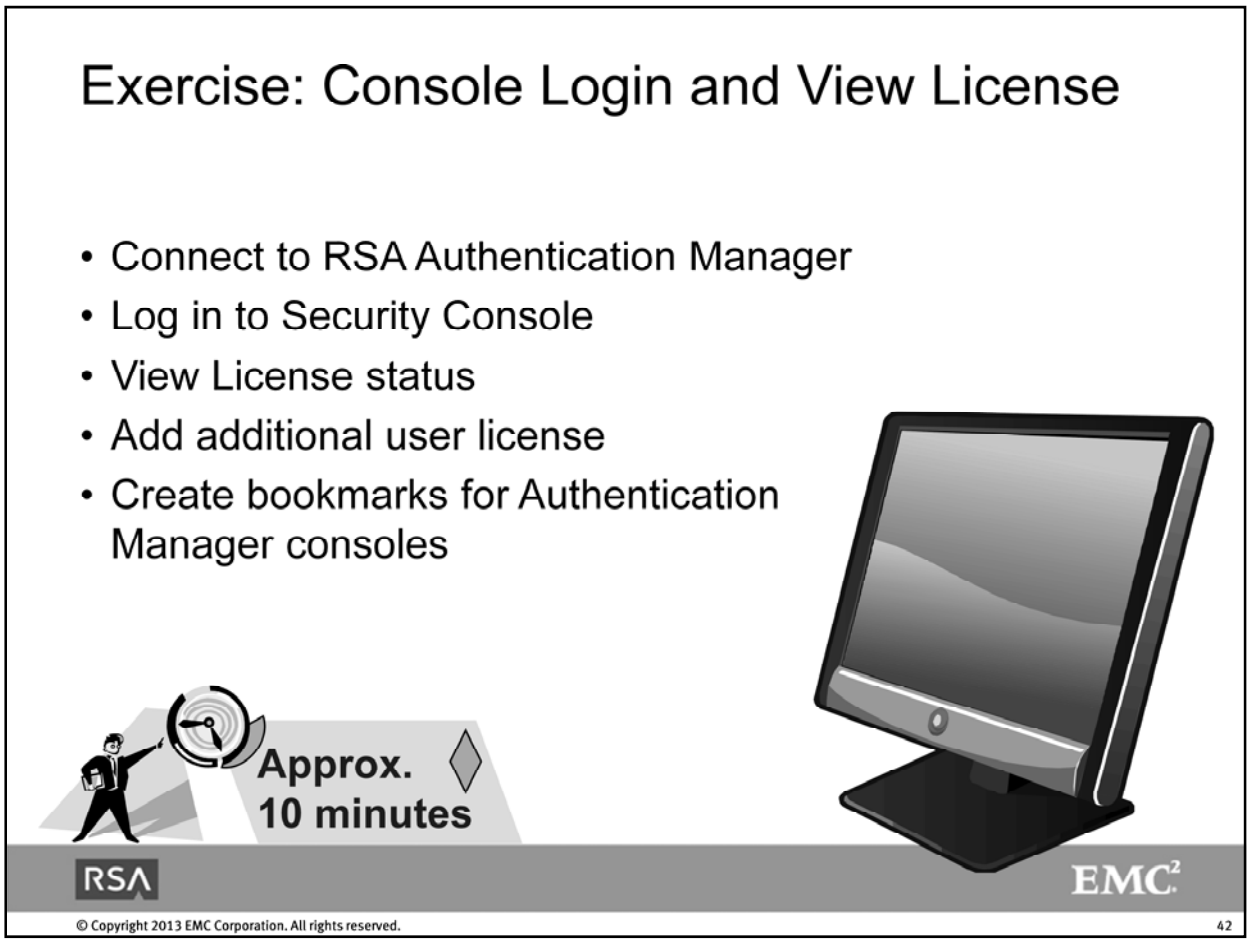
Business Continuity Option is designed to provide a rapid and short term 'boost' to the user limit of a license to accommodate emergency situations.

For example, if a natural disaster prevents users from physically going to the workplace, Business Continuity can allow an organization to assign credentials to a large number of individuals so they can access business resources remotely - as through a VPN or RSA SecurID-protected web portal. Such users could take advantage of authentication methods such as on-demand or Emergency Passcodes that would not require assignment of an RSA SecurID token.

Licensed Active Users

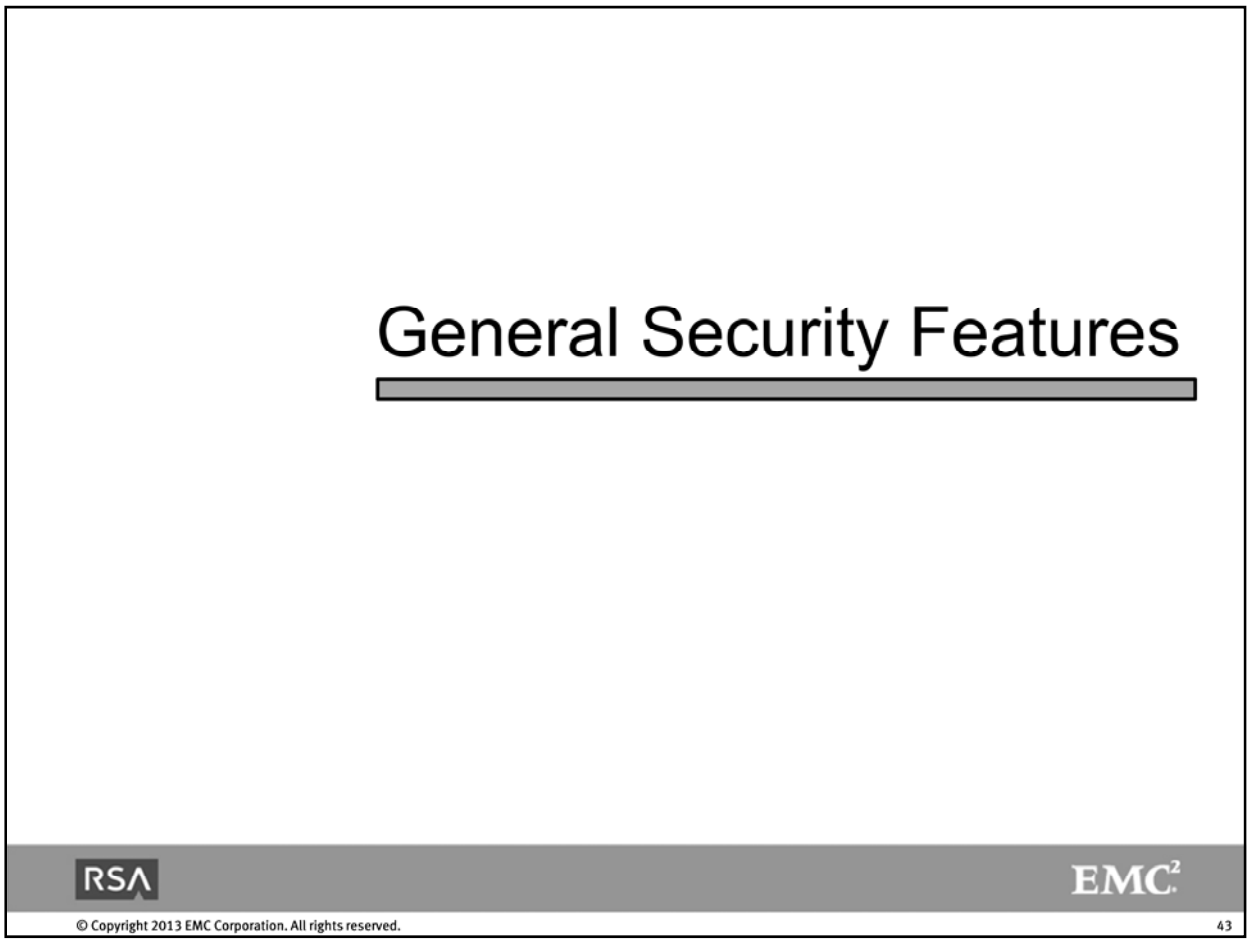
- Example; for a 10 user license:

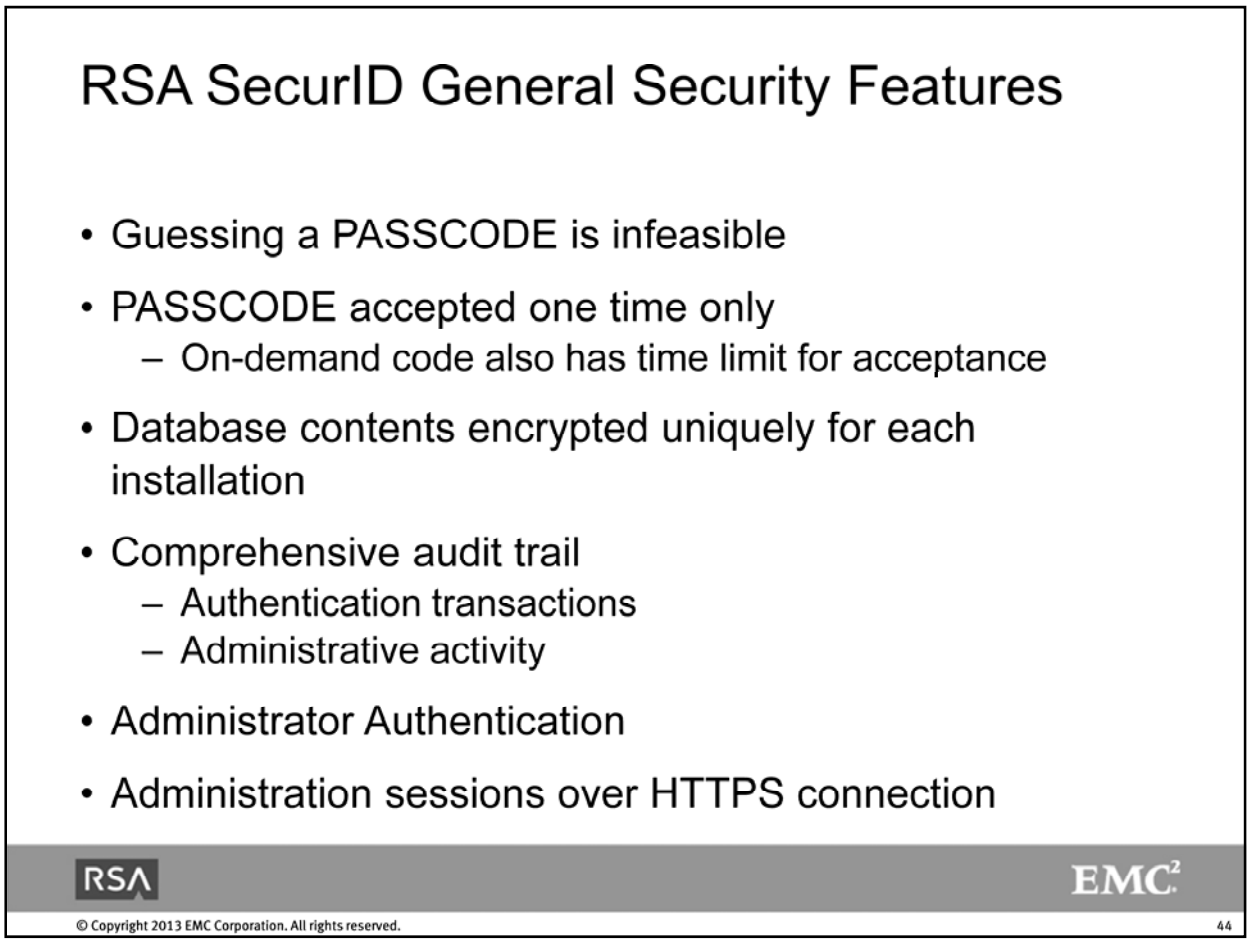




Console URLS:

- Security Console: https://<*Fully Qualified hostname or IP address*>:7004/console-ims
- Operations Console: https://< *Fully Qualified hostname or IP address* >:7072/operations-console
- Self-Service Console: https://< *Fully Qualified hostname or IP address* >:7004/console-selfservice


General Security Features
RSA
EMC²
© Copyright 2013 EMC Corporation. All rights reserved.
43



The RSA SecurID system has the following inherent security features to assist in evading a system attack:

Passcodes
The random nature of the RSA SecurID tokencode and the use of two factors to create a Passcode makes simple Passcode guessing infeasible.

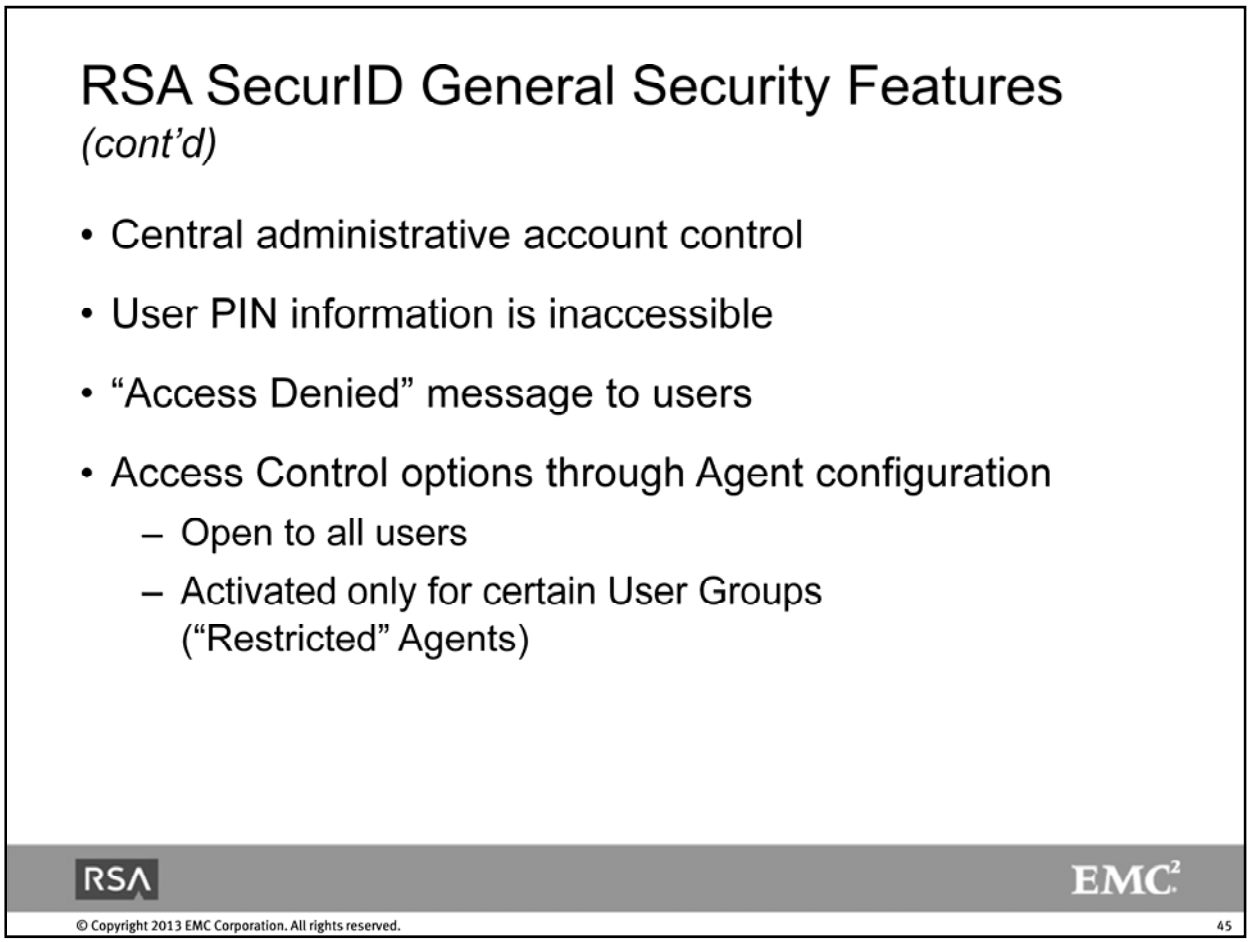
Single use of a Passcode
A Passcode can be used only once and cannot be reused. Any Passcode obtained through observation or electronic eavesdropping is worthless.

Database Encryption
The RSA Authentication Manager database is encrypted to prevent anyone from accessing confidential information.

Complete audit trail and user activity reporting
The RSA Authentication Manager logs all user authentication and administrative transactions.

Administrator authentication
Any administrator is required to authenticate before they are allowed access to the system. The administrator and security policy can dictate what authentication method or methods an administrator must use.

Secure Administrative sessions
Access to the administrative console is secured via an HTTPS connection from the browser to the web server.



Centralized Administration

The RSA Authentication Manager administration model offers one point of security administration.

PIN security

User PIN information is inaccessible to the administrator.

"Access Denied" Display

The RSA SecurID logon does not reveal information which might be of use to the would-be attacker.

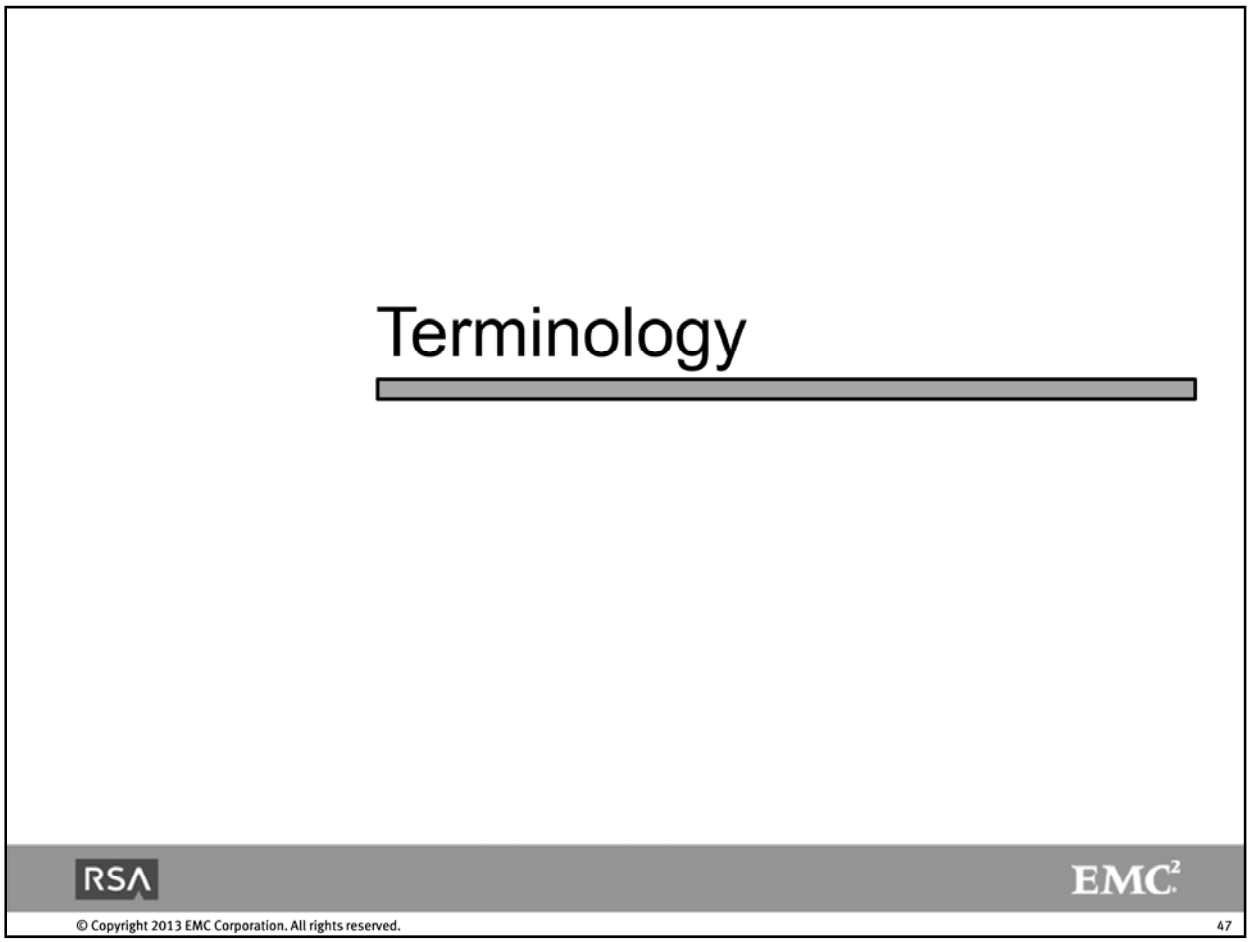
Agent access control (authorization / restriction)

Agent Hosts can be configured to allow authentication only to certain user groups. This allows you to control who is authorized to authenticate through (or restricted from) various network access points.

RSA SecurID General Security Features
(cont'd)

- Multiple authentication methods and form factors allow deployment flexibility
  - Can leverage a device that user already has and carries (e.g. phone)
  - On-demand authentication allows strong user authentication without token issue/deployment
  - Risk-based authentication allows setting authentication requirement based on risk level

RSA EMC²


Terminology
RSA
EMC²
© Copyright 2013 EMC Corporation. All rights reserved.
47

Terminology

- Deployment
  - All Authentication Manager hosts and entities across an enterprise
- Instance
  - Primary or Replica Authentication Manager server or RSA SecurID Appliance
- Primary [Replica] Instance
  - Virtual host (and usually associated physical host) that contains database, authentication services and administration applications
  - Read/Write access only to Primary instance (Replica is read-only)

RSA EMC²

Terminology *(cont'd)*

- Administrator
  - Authentication Manager user with one or more assigned administrative roles
- Agent
  - Software and/or device through which a user is stopped and prompted for authentication
- Authenticator (a.k.a. 'Token')
  - Device (e.g. Hardware Token) or method (e.g. On-Demand) by which a user can authenticate
- Security Domain
  - Organizational sub-division under a deployment

RSA EMC²

Terminology *(cont'd)*

- User
  - An account managed within Authentication Manager (usually a person but can also be a device or a service)
- User Group
  - A collection of users or user groups
- Member User Group
  - A user group nested within another user group
- Super Admin
  - Administrative role that has all permissions and scoping within a deployment
- Identity Source
  - Internal (Authentication Mgr database) or external (LDAP) store for user account data

RSA EMC²

Terminology *(cont'd)*

- On-Demand Authentication (ODA)
  - Authentication method where a user receives a code via email or SMS
- Risk-Based Authentication (RBA)
  - Authentication process where user characteristics (computer, login pattern, asset value, etc) determines the level of authentication required
- Provisioning
  - Process to assign and supply authentication device or method to an end user

RSA EMC²

51

Unit 1 Quiz

1. What is a Node Secret?
2. How does an Agent know how to send authentication requests to a Replica Server?
3. Which administrative function(s) can be performed on a Replica Server instance?
4. How many factors of authentication are 'secure'?
5. What database(s) are used to hold Authentication Manager user account information?

RSA SecurID Authentication

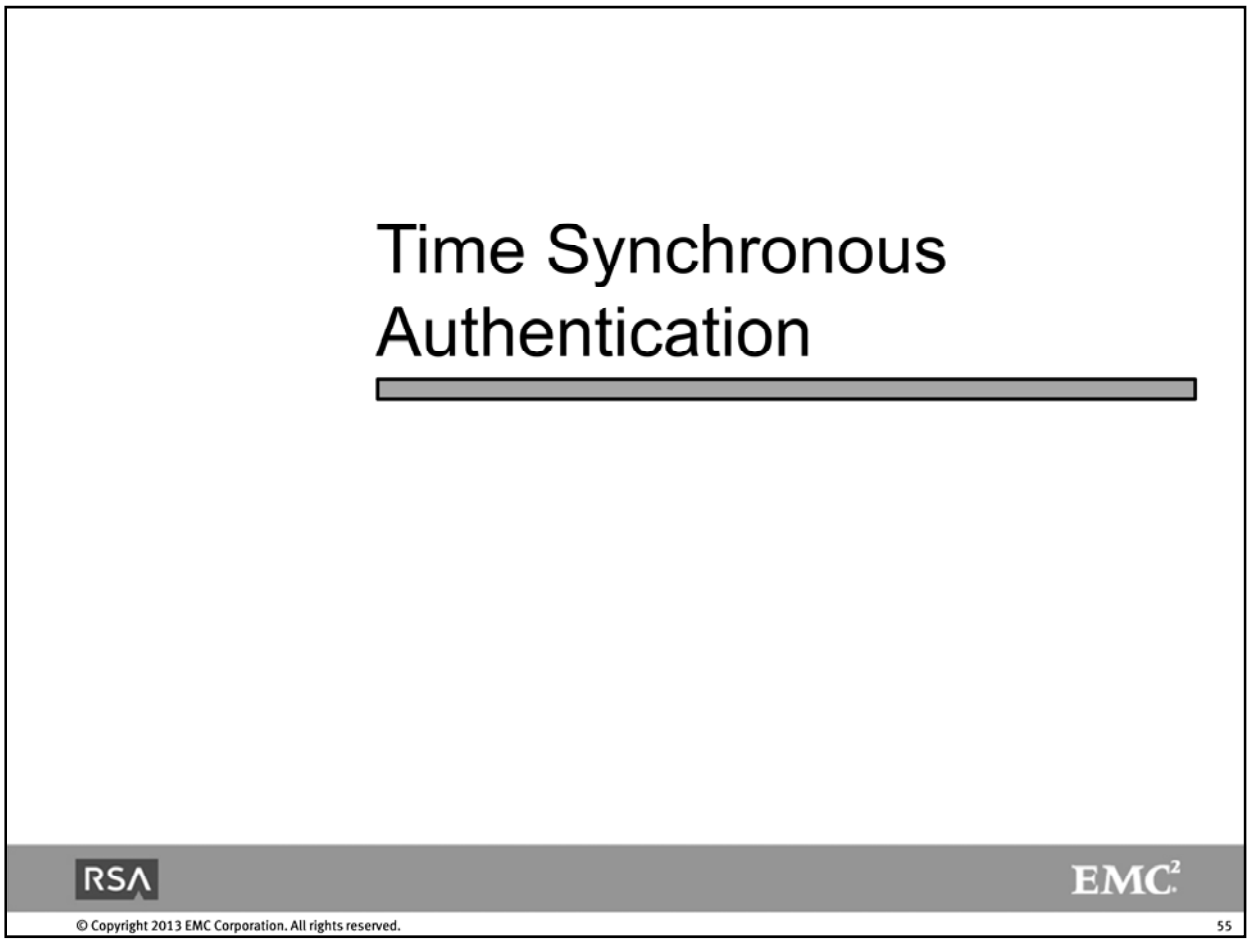
Unit 2

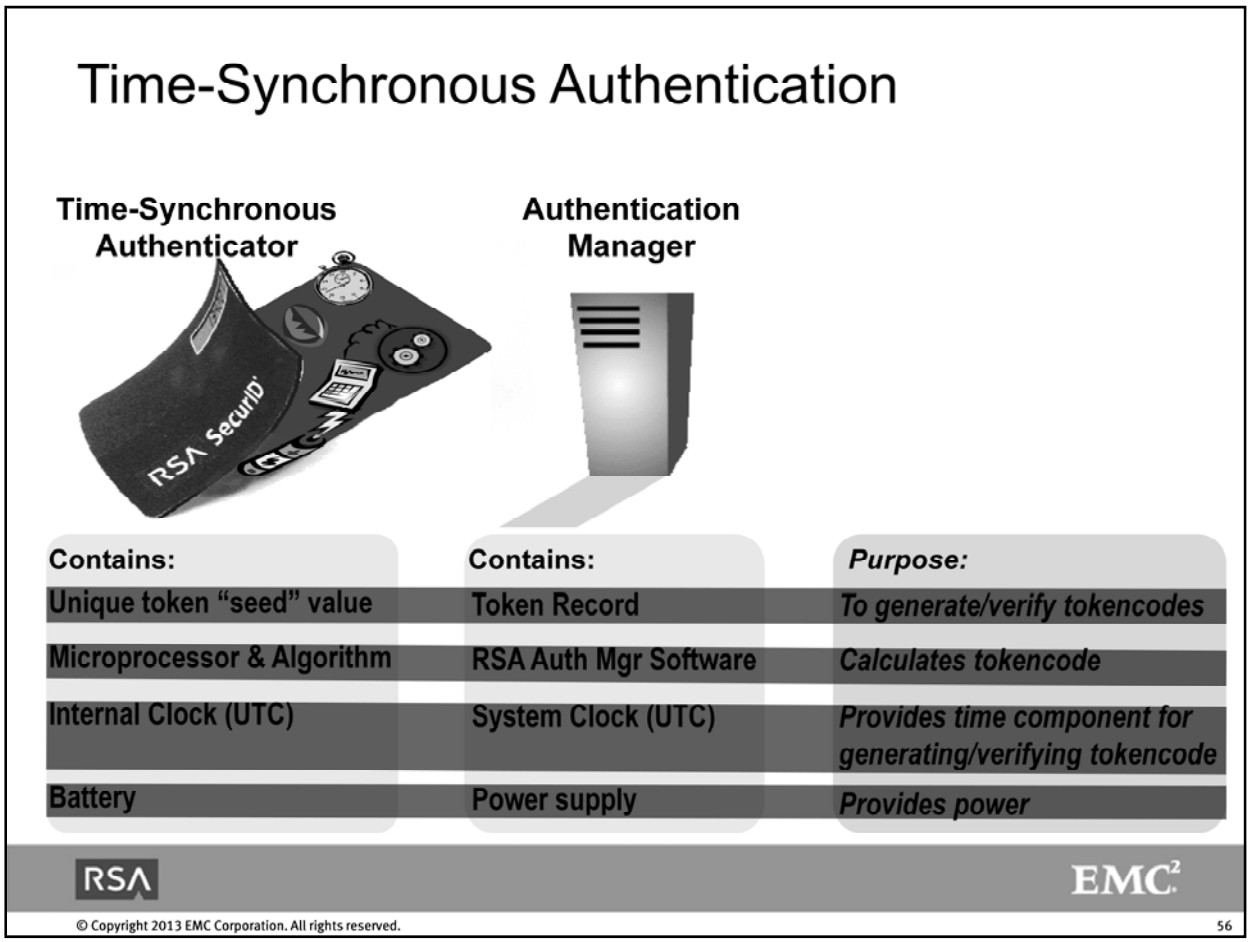
RSA

EMC²

53

RSA Security offers RSA SecurID Authenticators (“Tokens”) as hardware and software devices as well as offering strong authentication through On-Demand and Risk-Based Authentication.


Time Synchronous Authentication
RSA
EMC²
© Copyright 2013 EMC Corporation. All rights reserved.
55



RSA SecurID hardware and software tokens calculate and display a unique tokencode that changes at a specified interval – typically every 60 seconds.

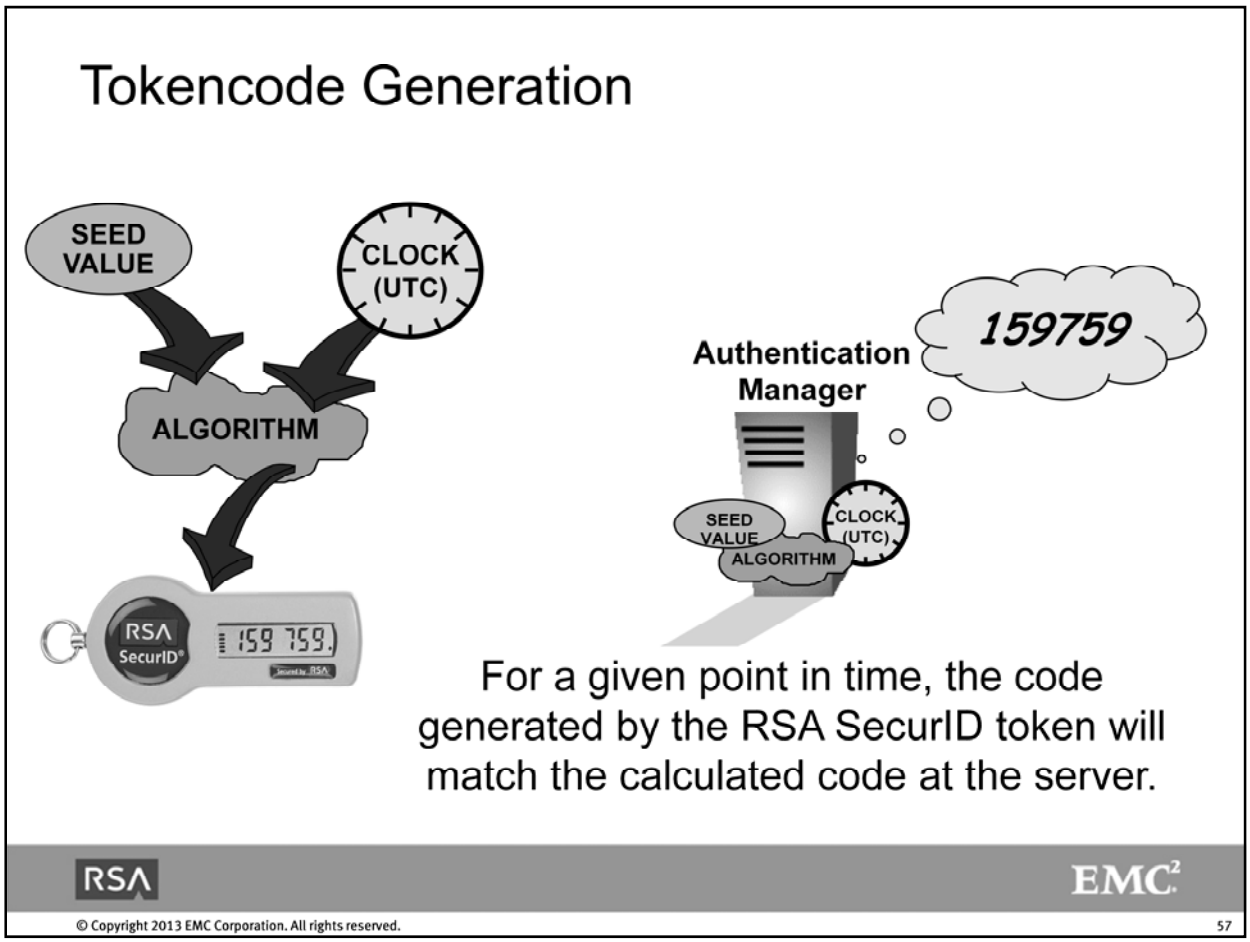
Time-Synchronous Authenticator Components:

- A seed value for pseudo-random number generation
- An algorithm with which to calculate tokencodes

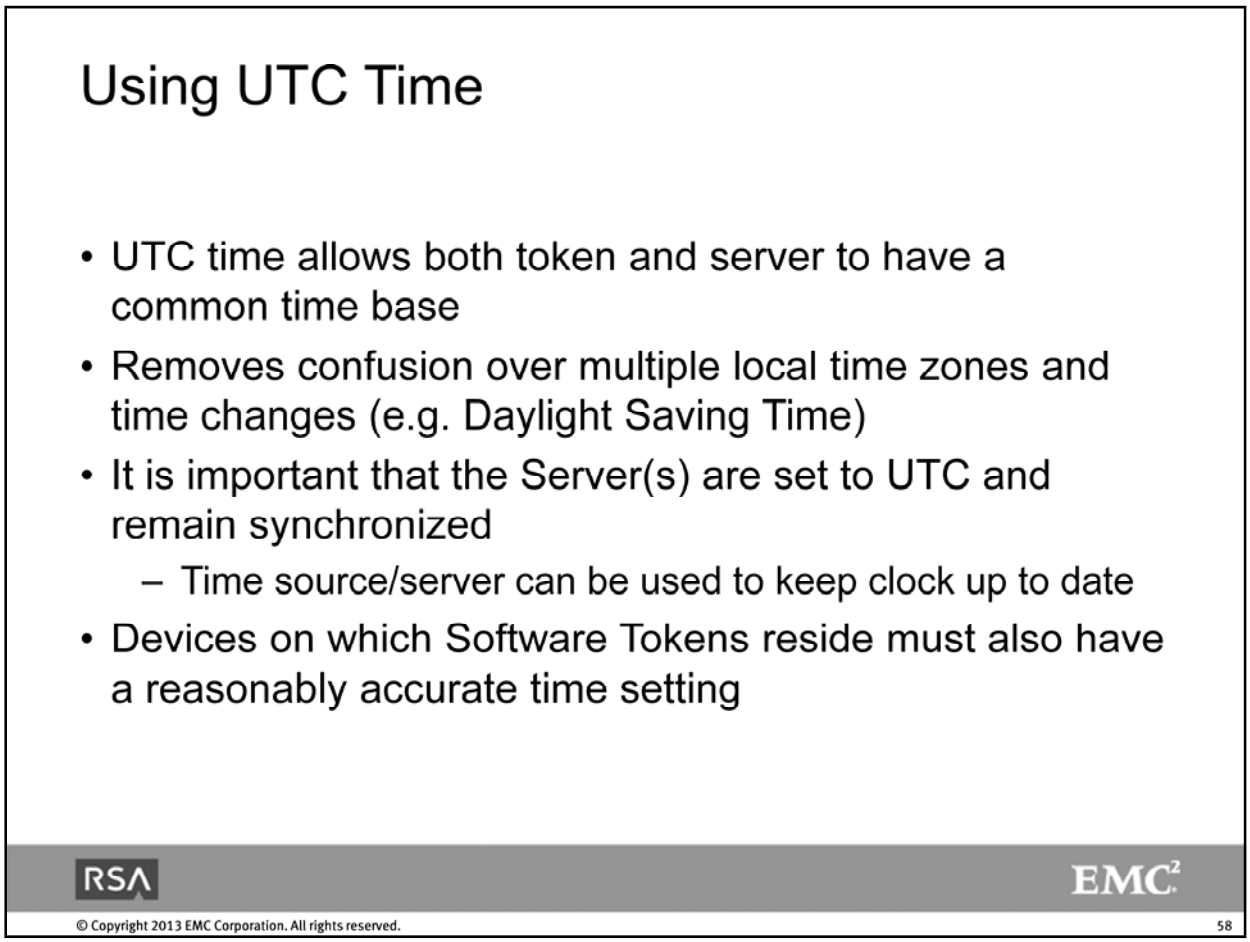
In addition, hardware tokens contain:

- A power source (battery)
- A microprocessor
- An accurate UTC clock

Software tokens utilize the power, processor, and clock from the host computing device on which they reside.

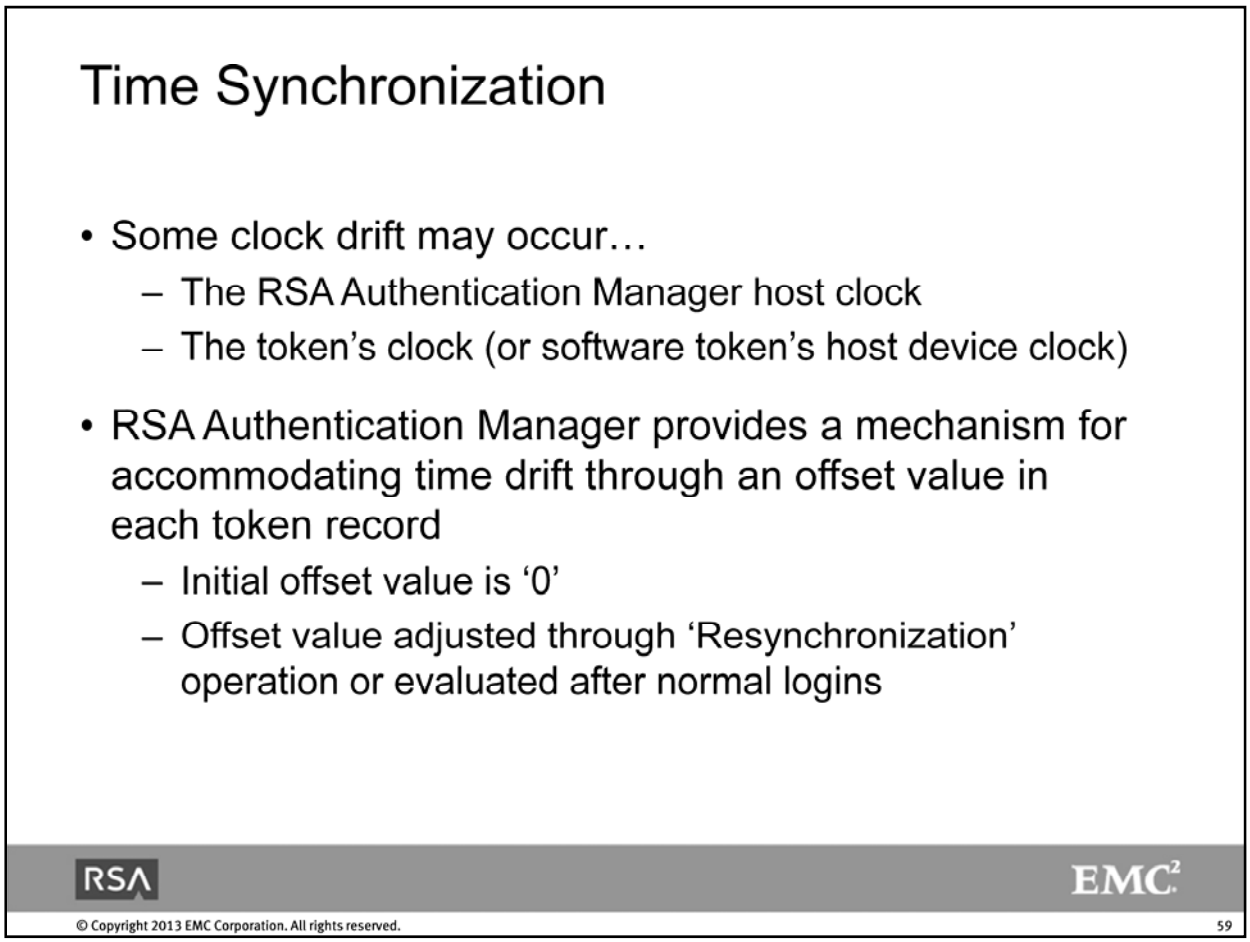


The RSA Authentication Manager and RSA SecurID time-synchronous tokens independently generate tokencodes based on the same algorithm. The tokencode produced by an RSA SecurID token and the tokencode calculated by the RSA Authentication Manager match at a given moment in time because both are basing the calculations on the same token "seed" record and time value - UTC (Universal Coordinated Time).



Universal Coordinated Time, UTC (the same as Greenwich Mean Time – GMT) is the basis for RSA SecurID time-synchronous tokencode generation. Time-synchronous hardware tokens are synchronized to UTC before being sealed in the factory. During RSA Authentication Manager installation, the Administrator is asked to synchronize the server host clock to UTC.

By using UTC, the issues associated with accommodating local time zones and adjustments for Daylight Savings Time are eliminated – every RSA SecurID token and RSA Authentication Manager in the world are initially synchronized to one standard time base.



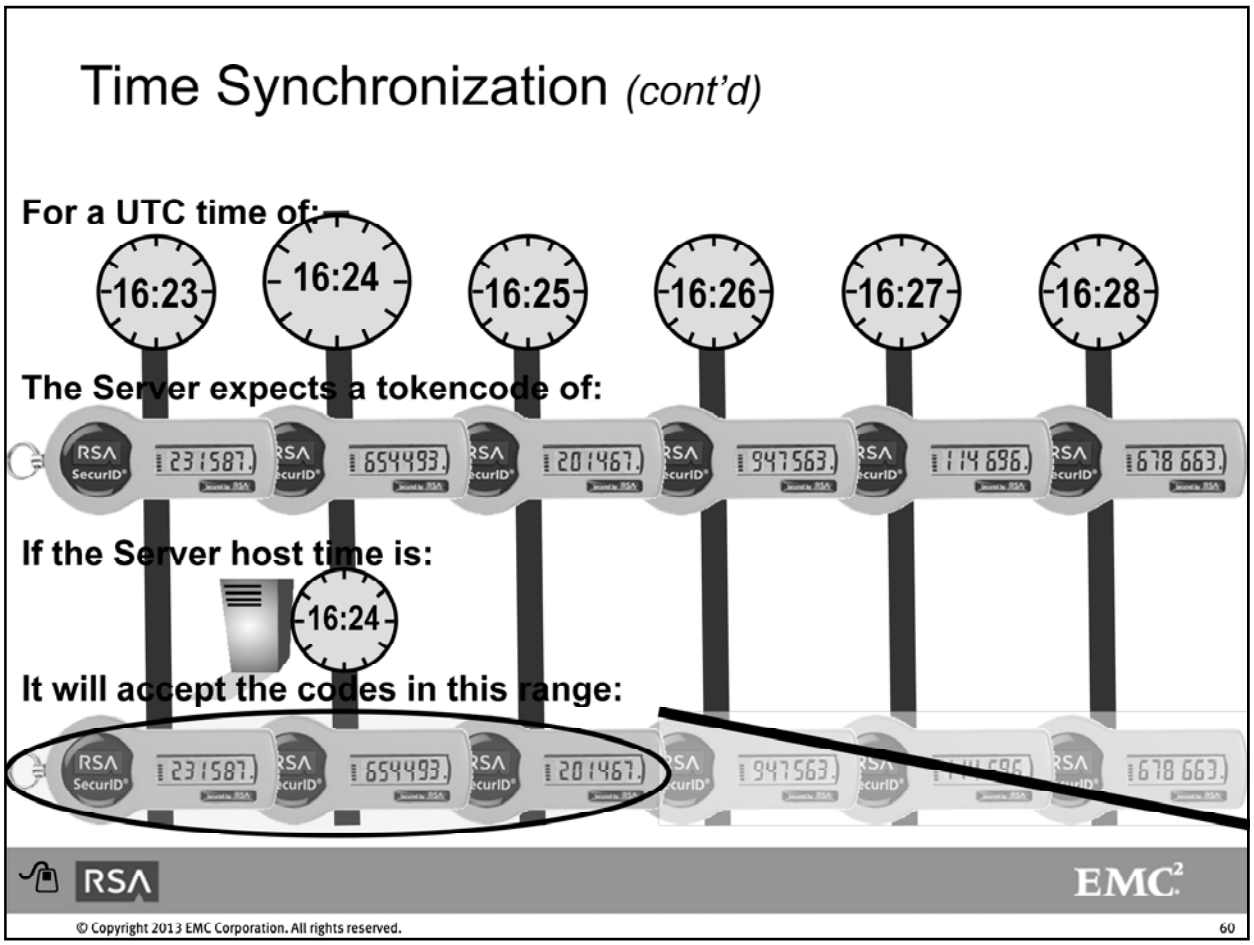
The RSA SecurID time-synchronous tokens and RSA Authentication Manager host clock are initially synchronized to UTC. Over time, the clock in either a token or in the RSA Authentication Manager software may drift slightly from the exact correct time. The RSA SecurID system allows some accommodation and adjustment for such drift.

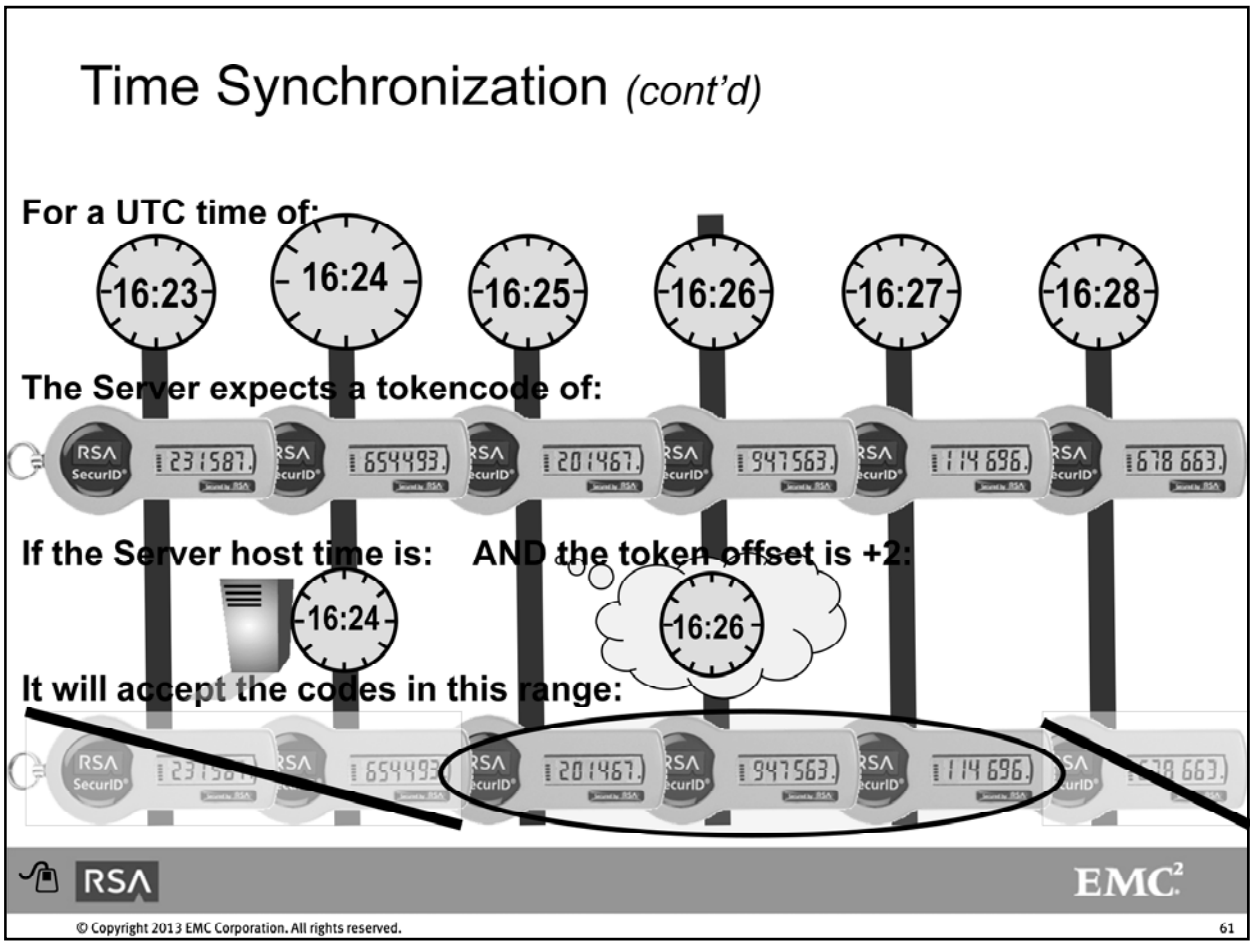
Every RSA SecurID token has an automatic adjustment time window of three intervals. For a token with a one-minute change interval, the range would represent three minutes.

A token calculates a tokencode based on the time of its internal clock. The clock may be using correct UTC time or may be drifting slightly fast or slow.

When a user enters a tokencode, the RSA Authentication Manager tries to verify that the tokencode is correct and if it is slightly ahead or behind it's target point, Authentication Manager will remember this variation and record it as an offset value.

At the next authentication, the offset value is applied to adjust the time componsent of the tokencode calculation.


Time Synchronization (cont'd)
For a UTC time of:
16:23
16:24
16:25
16:26
16:27
16:28
The Server expects a tokencode of:
231587
654493
201467
947563
114696
678663
If the Server host time is:
16:24
It will accept the codes in this range:
231587
654493
201467
947563
678663
RSA
EMC²
© Copyright 2013 EMC Corporation. All rights reserved.
60


Time Synchronization (cont'd)
For a UTC time of:
16:23
16:24
16:25
16:26
16:27
16:28
The Server expects a tokencode of:
231587
654493
201467
947563
114696
678663
If the Server host time is:
AND the token offset is +2:
16:24
16:26
It will accept the codes in this range:
RSA
EMC²
© Copyright 2013 EMC Corporation. All rights reserved.
61

Time Synchronization *(cont'd)*

If a code is *slightly* out of range, the system will prompt for a second code to verify the user has the correct token

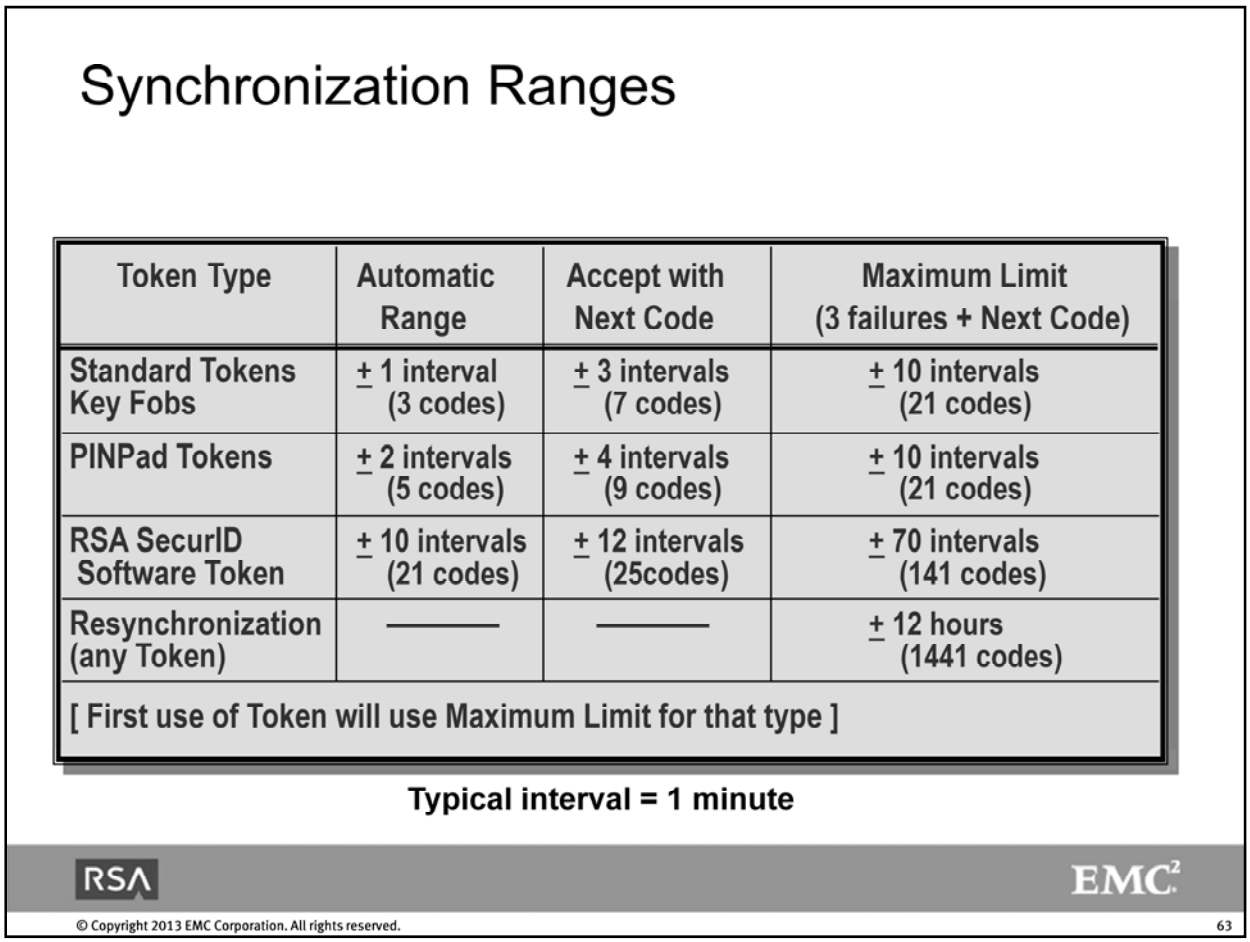
User initially uses this code

Receives a prompt for Next Tokencode

Waits for the code to change, then enters the next code

RSA EMC²

62



Different token types have different synchronization ranges. The ranges of acceptance for each type of Token are listed in this table.

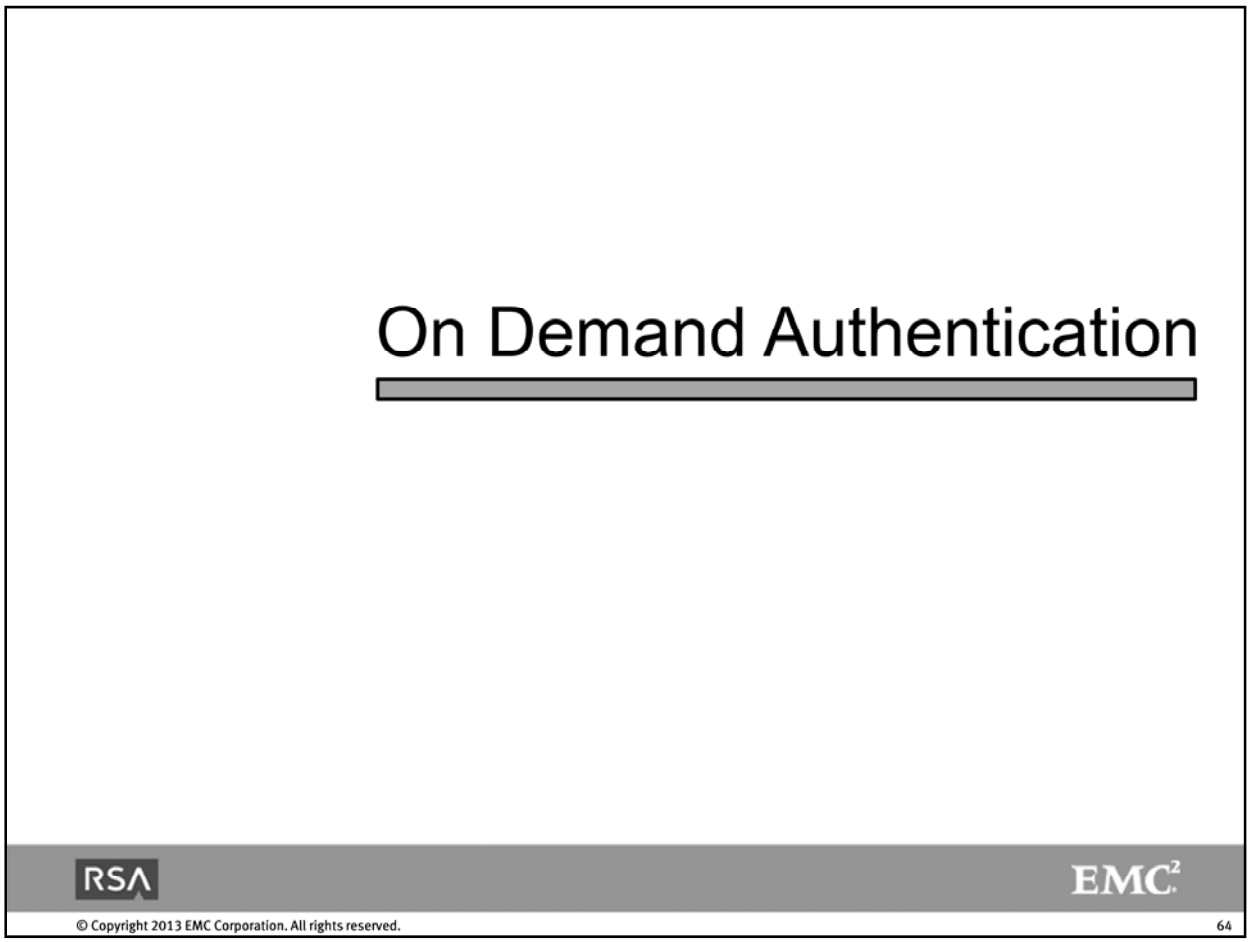
The following three ranges apply:

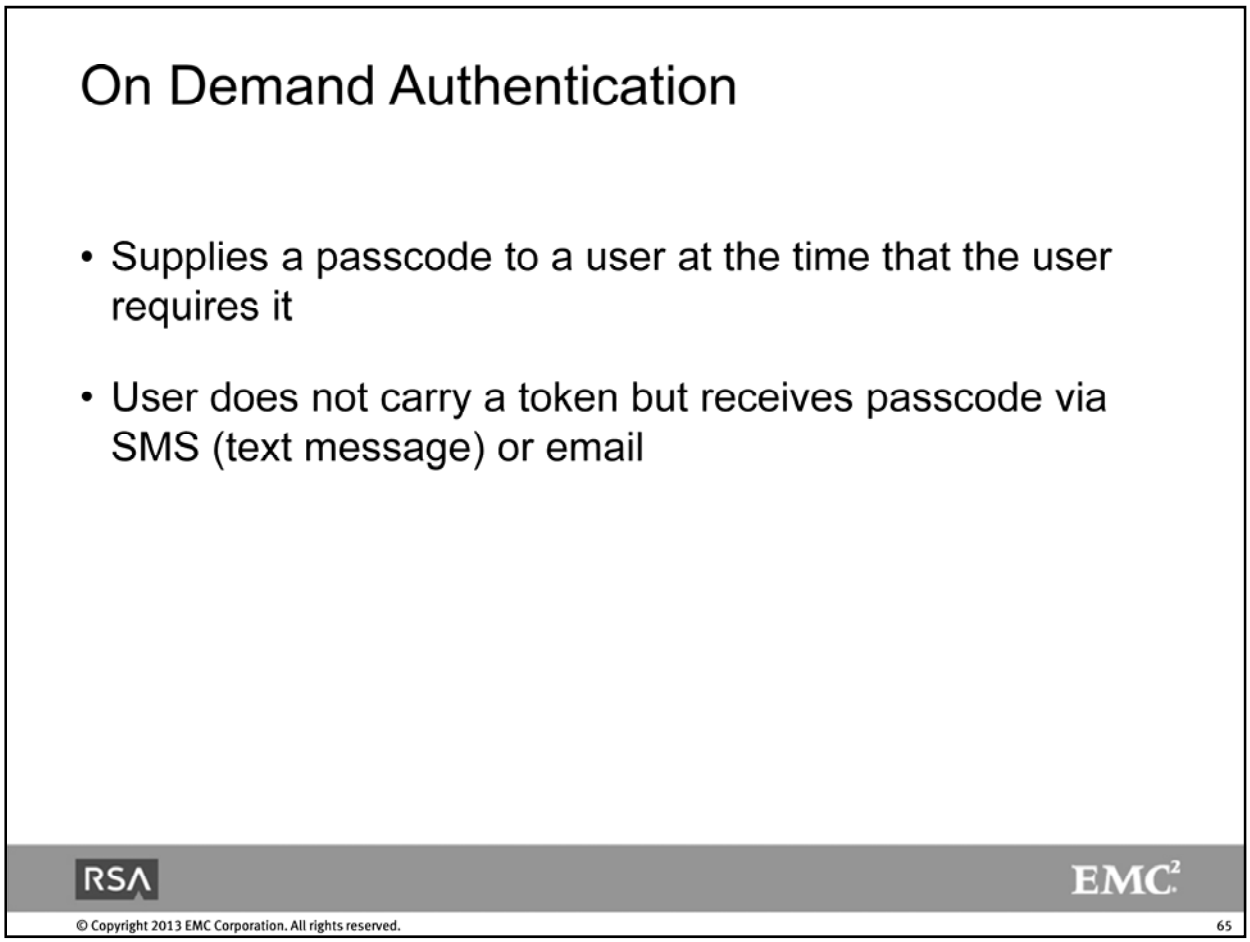
Automatic Acceptance Range: A narrow window within which only a relatively few tokencodes are acceptable.

Acceptance with Next Tokencode: A slightly larger window within which the first entered code is acceptable only if followed (when the user is prompted) by the next sequential tokencode.

Maximum Limit Range: A significantly larger window within which users will fail authentication ("Access Denied") for three attempts. Upon a fourth attempt, a larger number of tokencode possibilities are calculated by the RSA Authentication Manager (for example, 21 codes in the case of a Standard token). If the user's tokencode matches any one of these codes, a second, sequential tokencode is requested. If this second tokencode is confirmed, the user is granted access and the token record is updated with the new time offset value.

"Resynchronization" is an administrative function to allow Tokens which have drifted substantially to be brought back into a range acceptable to the RSA Authentication Manager.


On Demand Authentication
RSA
EMC²
© Copyright 2013 EMC Corporation. All rights reserved.
64



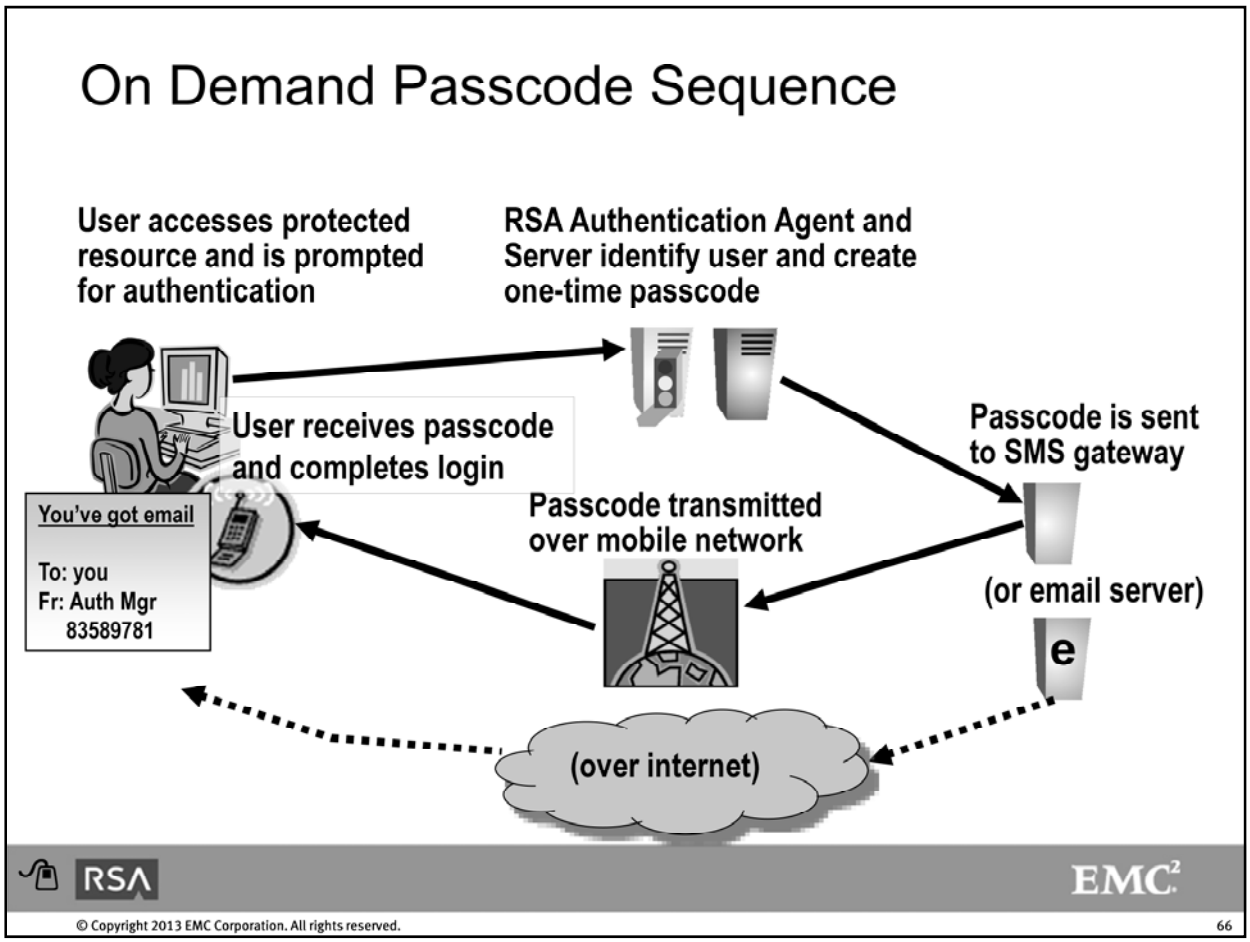
RSA Authentication Manager supports On-Demand Tokencodes through SMS (Short Message Service "Text Message") and SMTP (e-mail).

Default plug-ins and tokencode delivery information must be configured before using Authentication Manager to send on-demand tokencodes through the RSA Security Console.

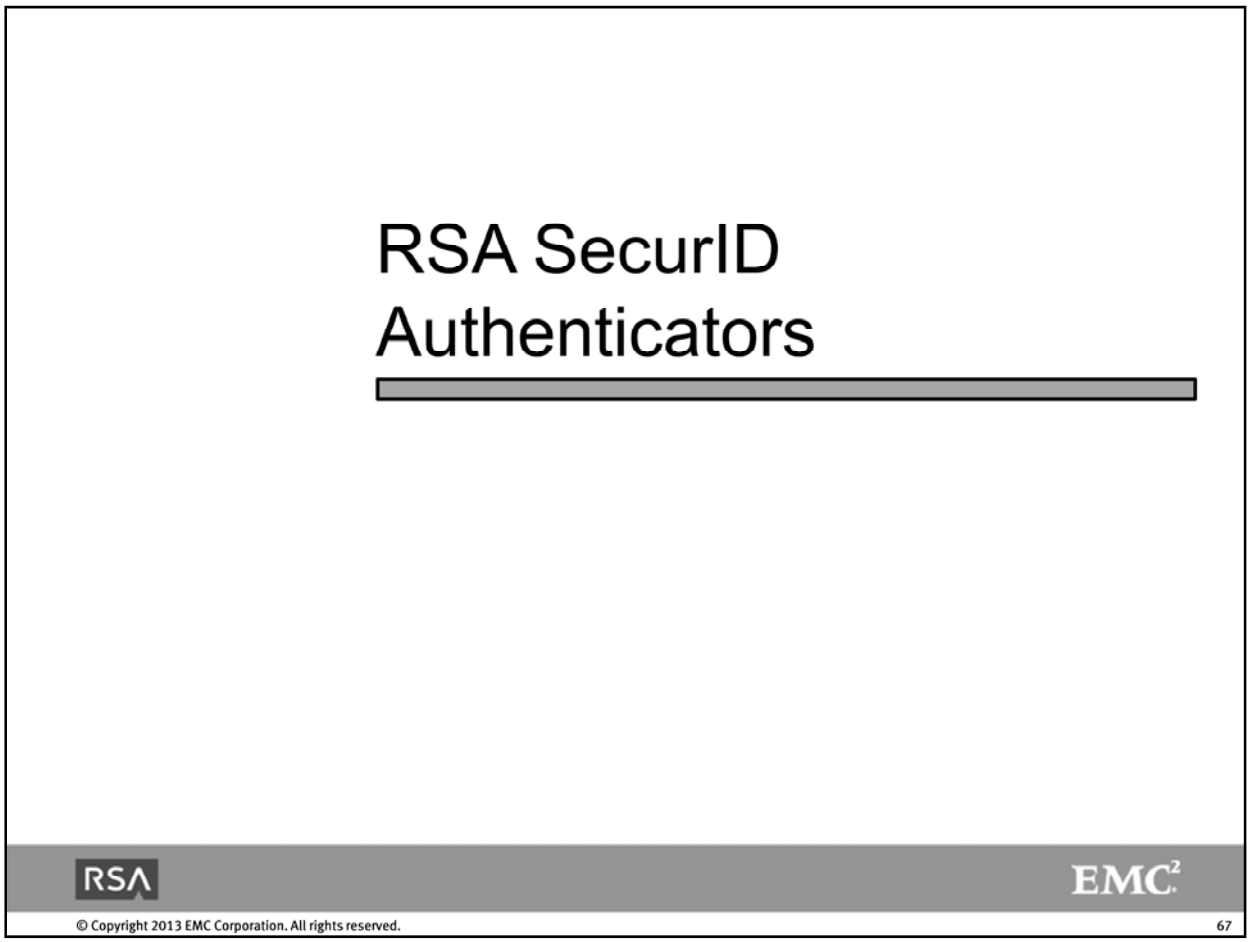
User data for SMS or email information can be assigned as an attribute that is mapped to either the internal Authentication Manager database or an external database (for example, telephoneNumber or emailAddress).

SMS Provider Information includes data such as API ID, account user name and password, and proxy server information. Authentication Manager uses plug-ins to integrate with Clickatell. If a different service provider is to be used, a new plug-in for that provider must be installed. To implement a custom plug-in for another provider, contact the RSA Professional Services Organization for more information.

Users must be enabled for SMS and SMTP authentication. Unlike a hardware or software token, an on-demand authenticator is not assigned. Instead, you enable the user to request and receive On-Demand tokencodes.



Not shown in this diagram is the use of a PIN. Typically, a user will need to supply a PIN before requesting an on-demand code.


RSA SecurID
Authenticators
RSA
EMC²
© Copyright 2013 EMC Corporation. All rights reserved.
67

Hardware Authenticators (Tokens)

Strength of Security	2 factors – PIN + tokencode
Typical Use	Mobile employee access
Client side Requirements	None
Portability	Works anywhere
Multiple use	No
User Challenges	Minimal
Distribution Requirements	Assign & Deliver tokens
System Requirements	Authentication server and Agents

Key fob
(SD700)

Standard card
(SD200)

Can use Alpha-numeric PIN

PINpad card
(SD520)

PIN must be numeric; no leading '0'

RSA EMC²

68

Hybrid Tokens

Strength of Security	2 factors – PIN + tokencode or certificate
Typical Use	Internal users or Mobile employees
Client side Requirements	Middleware for connected features
Portability	Passcode works anywhere, connection requires USB
Multiple use	File/mail encryption, Digital signing & remote access
User Challenges	Minimal
Distribution Requirements	Assign & Deliver token, Client software, Certificate delivery
System Requirements	Authentication server, Cert. Authority, Agents

Combined One Time Passcode with smartchip capability
(SID800)

- **Multiple X.509 certificates**
- **Multiple UserID/Password combinations**
- **Applications can access credentials programmatically**

RSA EMC²

69

Software Tokens

Strength of Security	2 factors – PIN + tokencode
Typical Use	Mobile employee access
Client side Requirements	Compatible PC/Device
Portability	Works on assigned system
Multiple use	No
User Challenges	Minimal
Distribution Requirements	Assign & Deliver software & seeds
System Requirements	Authentication server and Agents

(SD820)

PIN must be numeric; no leading '0'

Software Tokens On a USB Device

Strength of Security	2 factors (can be biometric protected)
Typical Use	Mobile employee access
Client side Requirements	Compatible USB device
Portability	Works anywhere – needs USB port
Multiple use	File storage
User Challenges	Minimal
Distribution Requirements	Assign & Deliver software & seeds
System Requirements	Authentication server and Agents

RSA EMC²

71

Software Token On a Mobile Device

Strength of Security	2 factors – PIN + tokencode
Typical Use	Mobile employee access
Client side Requirements	Compatible platform
Portability	Works anywhere
Multiple use	General purpose device
User Challenges	Minimal
Distribution Requirements	Assign & Deliver software & seeds
System Requirements	Authentication server and Agents

RSA EMC²

72

On-Demand Code

Strength of Security	2 factors – PIN + code
Typical Use	Occasional / Temp users Emergency access
Client side Requirements	Device capable of receiving Email or SMS
Portability	Works within service coverage area
Multiple use	No
User Challenges	Two step process
Distribution Requirements	None
System Requirements	Authentication server, Agent & SMS/email delivery method

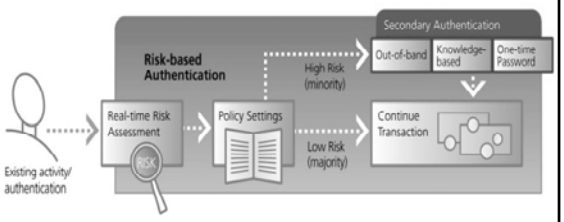
RSA EMC²

73

Choices: Risk-based Authentication

Strength of Security	2 + factors depending on risk assessment
Typical Use	Large consumer deployments – SSL VPN
Client side Requirements	None
Portability	Browser-based application
Multiple use	No
User Challenges	Minimal to difficult
Distribution Requirements	None
System Requirements	Authentication server, Agents, Web-based apps

- **Password or PIN + Invisible Device Authentication**
- **and/or Behavioral Profiling**
  - **What are you doing?**
  - **When are you doing it?**
  - **From where are you doing it?**
  - **Is this expected behavior?**



RSA EMC²

74

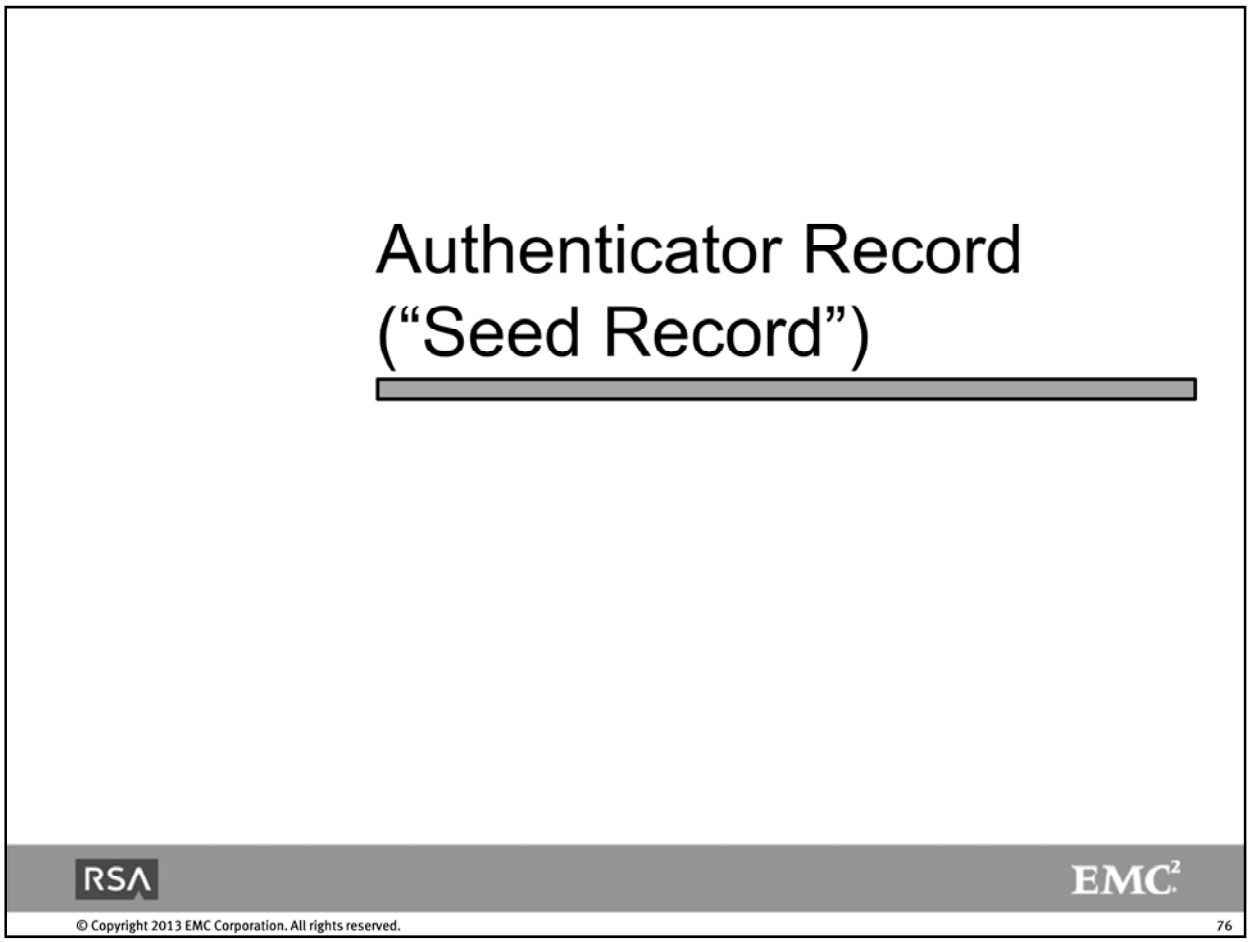
Security Questions

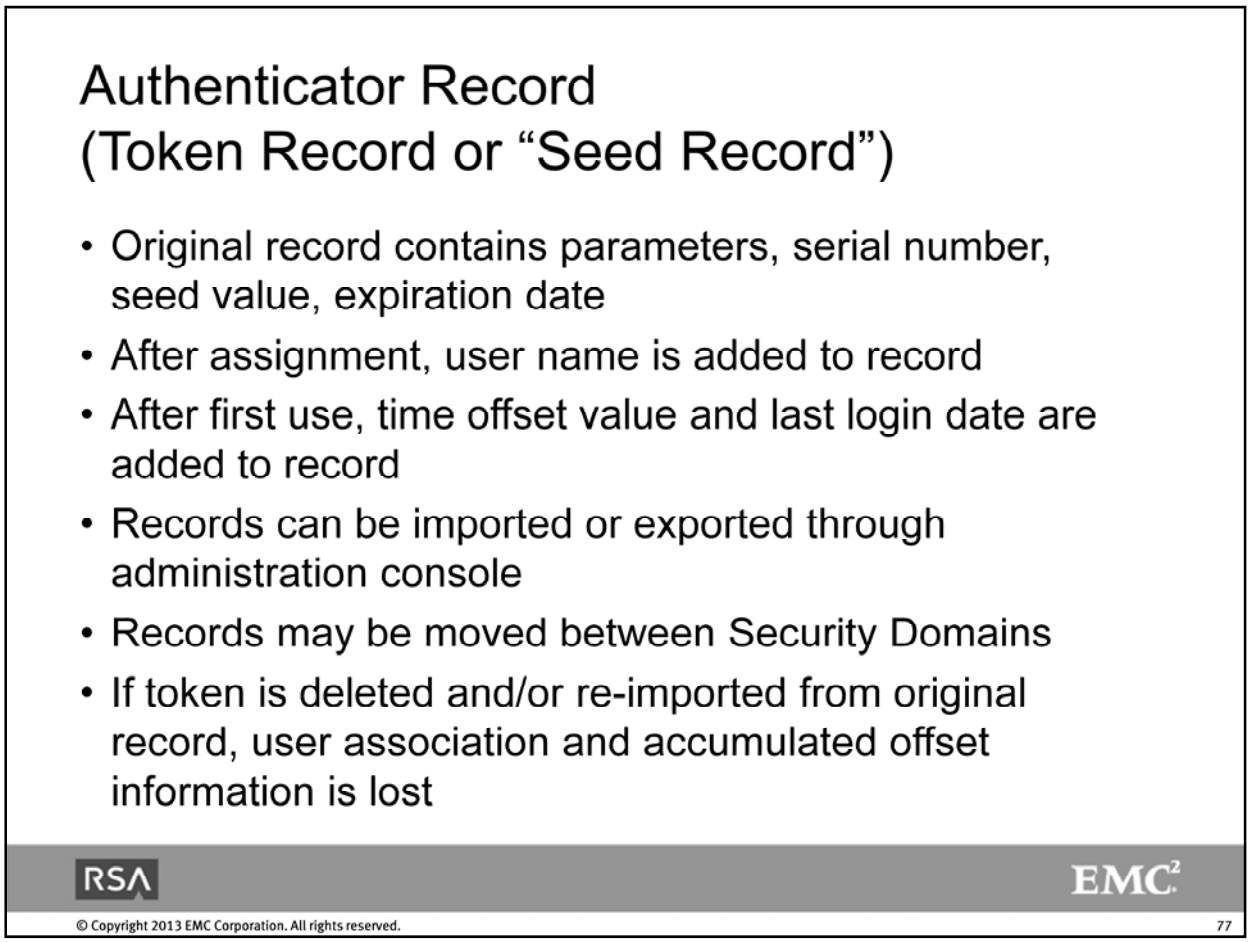
Strength of Security	1+ factors (***m*** of ***n*** of questions)
Typical Use	New user enrollment Emergency access/PIN reset
Client side Requirements	None
Portability	Works anywhere
Multiple use	No
User Challenges	Minimal
Distribution Requirements	None
System Requirements	Internet connection

Used for enrollment with user Self-service

RSA EMC²

75


Authenticator Record
("Seed Record")
RSA
EMC²
© Copyright 2013 EMC Corporation. All rights reserved.
76



Each RSA Security authenticator has a unique token record (informally called a "seed" record). A token record contains all data associated with that particular token and is used by Authentication Manager to calculate tokencodes.

Once a token begins to be used, the record accumulates information used to adjust for time drift or, in the case of event synchronous tokens, the sequence value of the most recently used code.

Token records must be imported into the Authentication Manager database before a token can be assigned to a user.

Records can be moved from one Security Domain to another to facilitate administrative access to an appropriate set of tokens for a set of users but records cannot be moved between Realms. In the case of Realms, tokens must be imported separately into each individual Realm.

One of the specifications, which is part of the token record, is the token's expiration date. The expiration date is defined by the length of token life ordered by a customer. An administrator should be aware of token expiration dates and make the necessary arrangements for replacement prior to expiration.

Authenticator Record Distribution

- Physical RSA SecurID tokens are supplied to a customer through shipment from RSA or RSA authorized reseller/distributor
- Token record media may be supplied separately
- Import Password is shipped separately
- Token records are unlocked by password when imported into Authentication Manager

Exercise: Import Token Records

- Log in to Security Console
- Locate and import Token Records
- "Receive" new tokens and records and import them

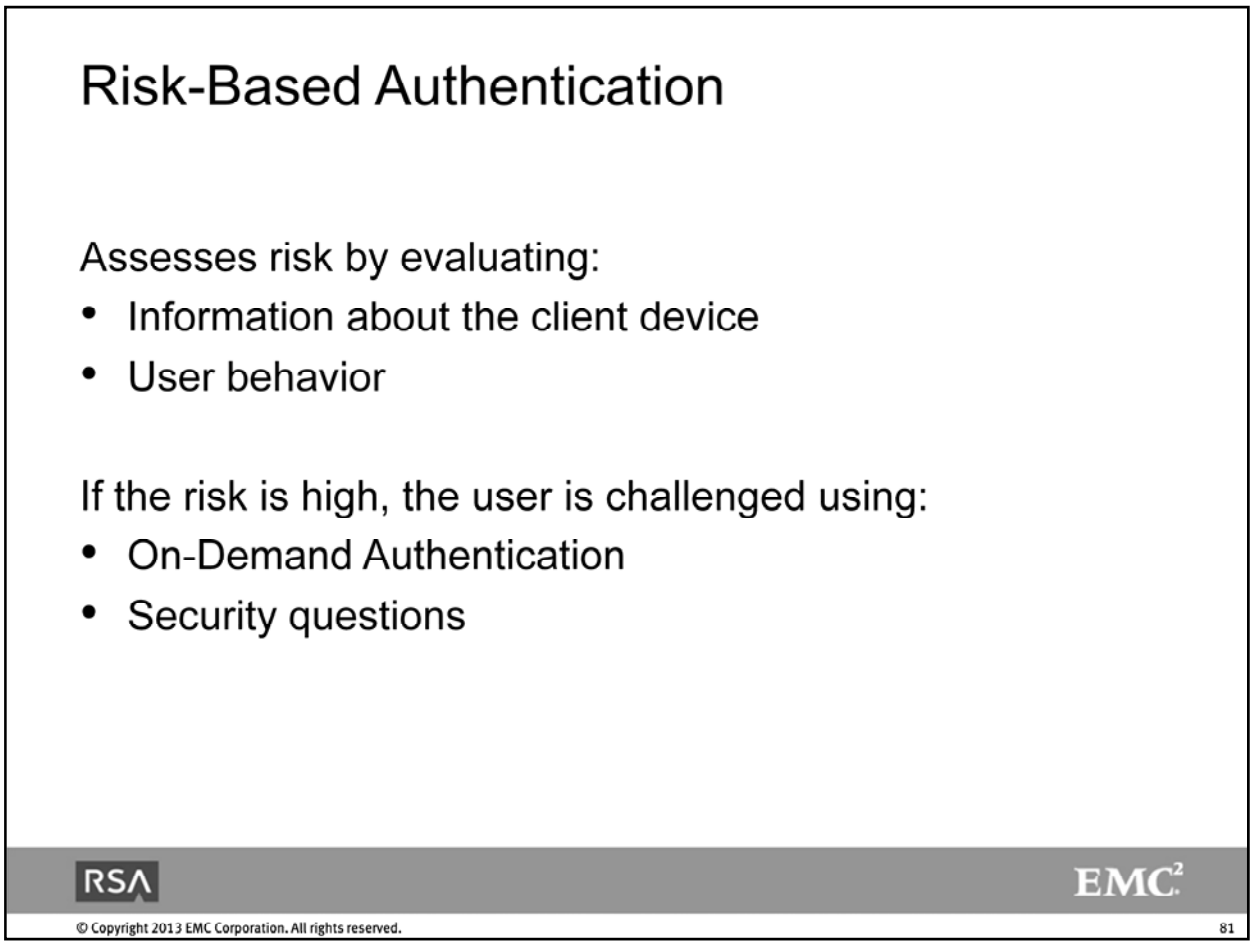
RSA EMC²

Risk-Based Authentication

Unit 3

RSA EMC²

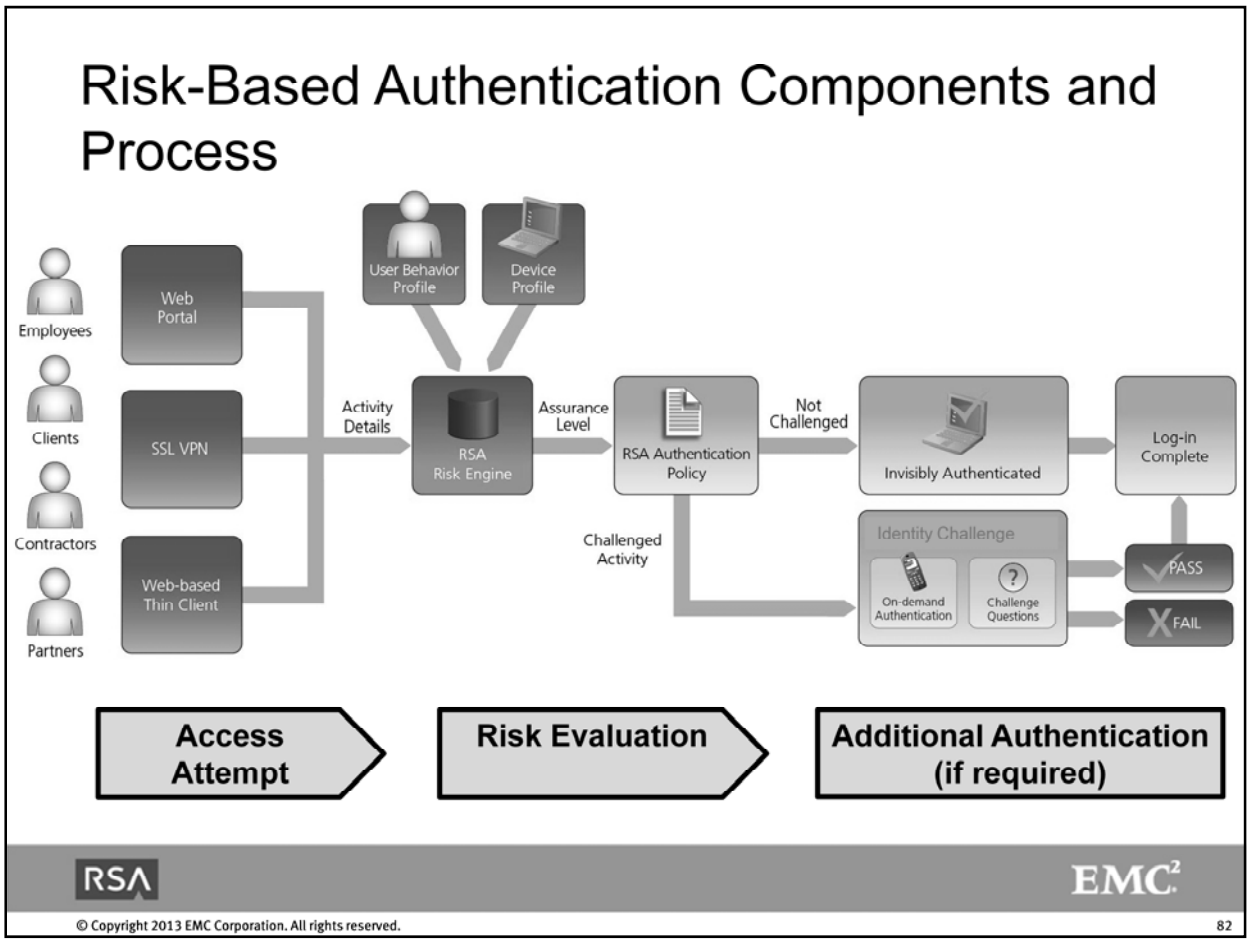
80



Risk-Based Authentication (RBA) is an authentication solution that strengthens traditional password based systems by incorporating knowledge of the client device and user behavior to assess the potential risk of an authentication request.

If the assessed risk is high, the user is challenged to further confirm his or her identity using one of the following methods:

- On-demand authentication. The user must correctly enter a PIN and a one-time tokencode that is sent to a preconfigured mobile phone number or e-mail account.
- Security questions. The user must correctly answer one or more pre-enrolled security questions.



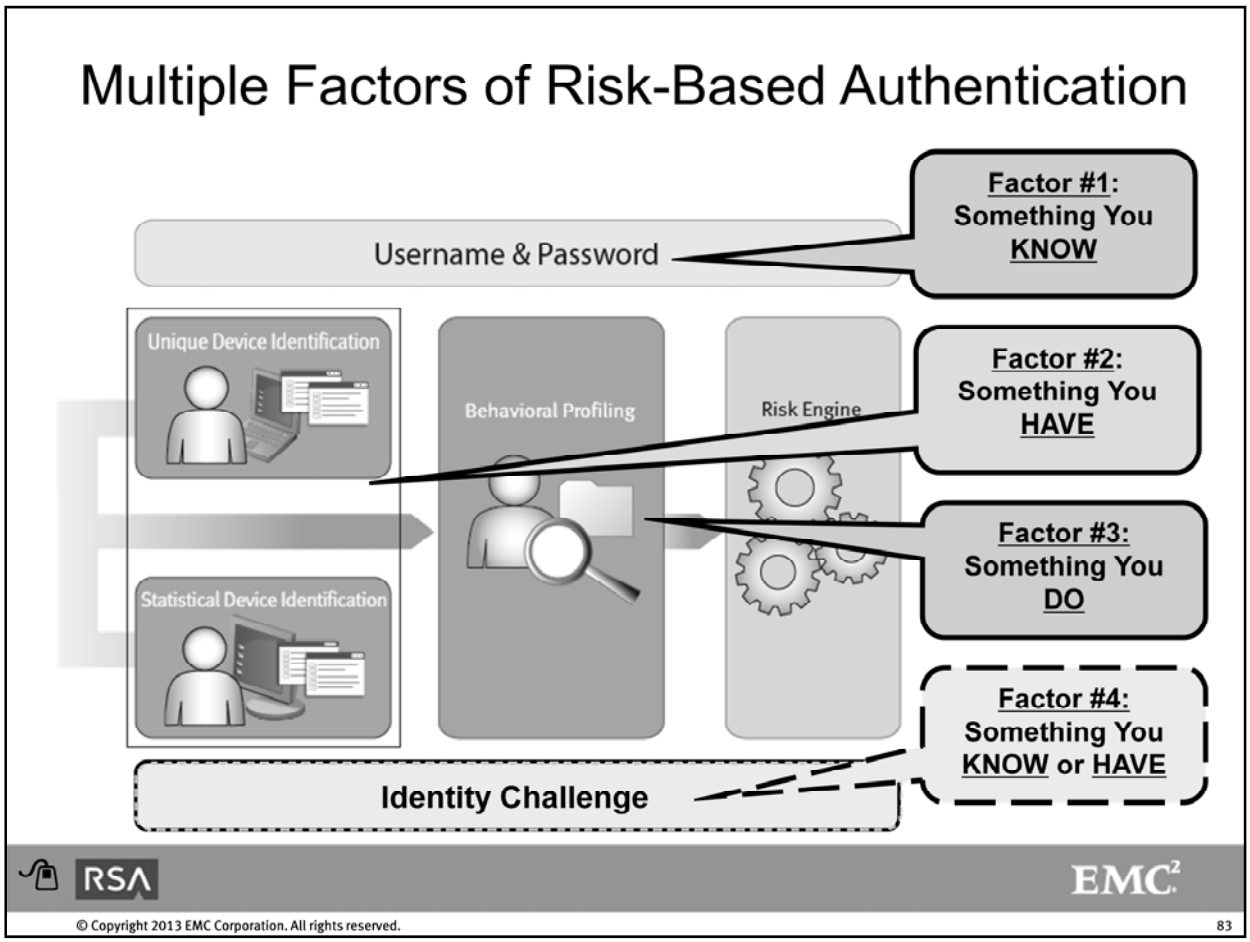
This flow diagram demonstrates the mechanics of risk-based authentication.

First, an employee, client, contractor, partner, or other individual attempts to access a system, such as a web portal, SSL VPN or web-based thin client protected by Risk-based authentication.

When a user authenticates using a username and password, the RSA Risk Engine makes an assessment and provides an assurance level. The two main categories of information analyzed are the User Behavior Profile and the Device Profile. The RSA Risk Engine evaluates each authentication attempt, tracking dozens of indicators and generating a unique risk score, which maps to an assurance level.

The assurance level is then compared to the Authentication Policy, established by the organization based on the organization's tolerance for risk and profile of users. An authentication attempt with a calculated Assurance Level that meets the minimum allowable limit, allows the user to continue on as normal without any change to his or her user experience. Thus, there is no impact on usability.

Conversely, if the assurance is below the minimum required, the user will be challenged for additional proof of identity. This is called identity challenge. If the end-user successfully completes the identity challenge, either by entering an OTP code delivered via SMS or e-mail, or answering security questions, then the login is completed. If the end-user fails the identity challenge, then the user is not permitted access to the resource.



This illustration shows how RSA Risk-Based Authentication is multi-factor.

Risk-based authentication takes advantage of three factors:

1). Something the user knows (the existing username & password)

2). Something the user has (a laptop or desktop device)

3). Something the user does (behavioral patterning analysis)

In situations during which the RSA Risk Engine determines an activity is high risk, an additional level of authentication is required in the form of a repetition of one of the three factors listed. In Authentication Manager Express, the identity challenge authentication methods are either something the user knows, for example, answers to security questions, or something the user has, such as a tokencode, delivered by out-of-band SMS to a mobile device in the user's possession.

The Risk Engine

The Authentication Manager risk engine is the same as used in RSA Adaptive Authentication to secure more than 250 million online identities.

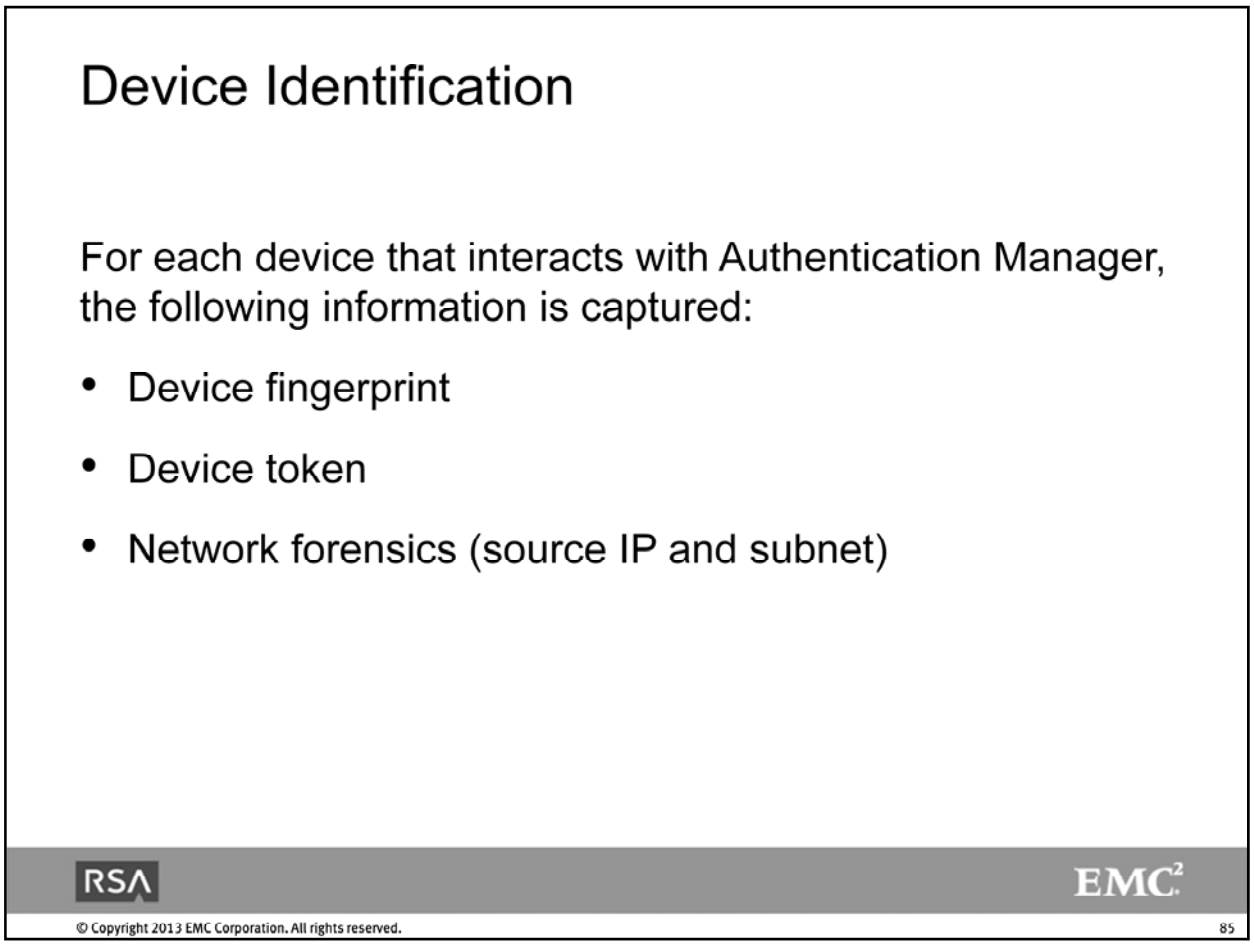
- A self learning engine that continually updates its risk model
- Factors used to determine a risk score include:
  - Device matching
    - Devices bound to the user account are low risk
    - Positive ID = high assurance + lower risk
  - Behavioral anomalies
    - Lowers overall assurance & Raises risk level
  - Overall assurance = device match + user behavior

RSA EMC²

84

The RSA risk engine takes advantage of technology used in RSA Adaptive Authentication to secure more than 250 million online identities.

It is a self-learning engine that constantly updates its risk model based on the behavior of individual users and your entire user population.



Device information is a collection of facts about a user's machine. These collected facts are evaluated by the risk engine to help identify fraudulent authentication attempts.

For each device that interacts with Auth Manager, the following information is captured:

- Device fingerprint
- Device token
- Network forensics (source IP and subnet)

If the device can be identified as a registered, or bound device for that user, the authentication attempt is considered low risk; otherwise, the user is considered a higher risk and will be challenged.

Device Identification – Device Fingerprint

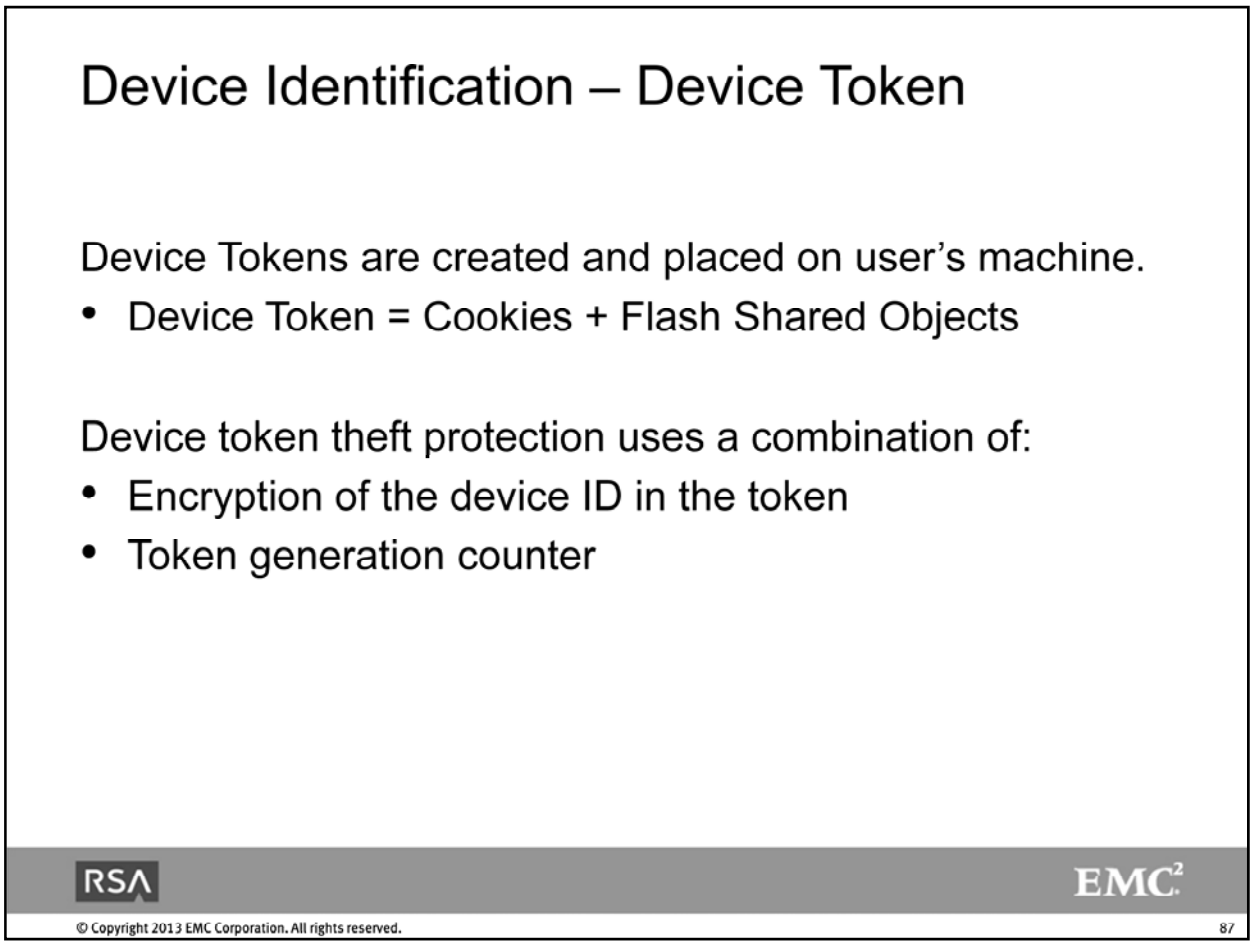
Fingerprint attributes include:

- User agent string
- System display
- Software fingerprint
- Browser language
- Time zone
- Language
- Cookies
- Java-enabled

RSA EMC²

86

To create a device fingerprint, the risk engine analyzes the detailed hardware and software characteristics of each computer.



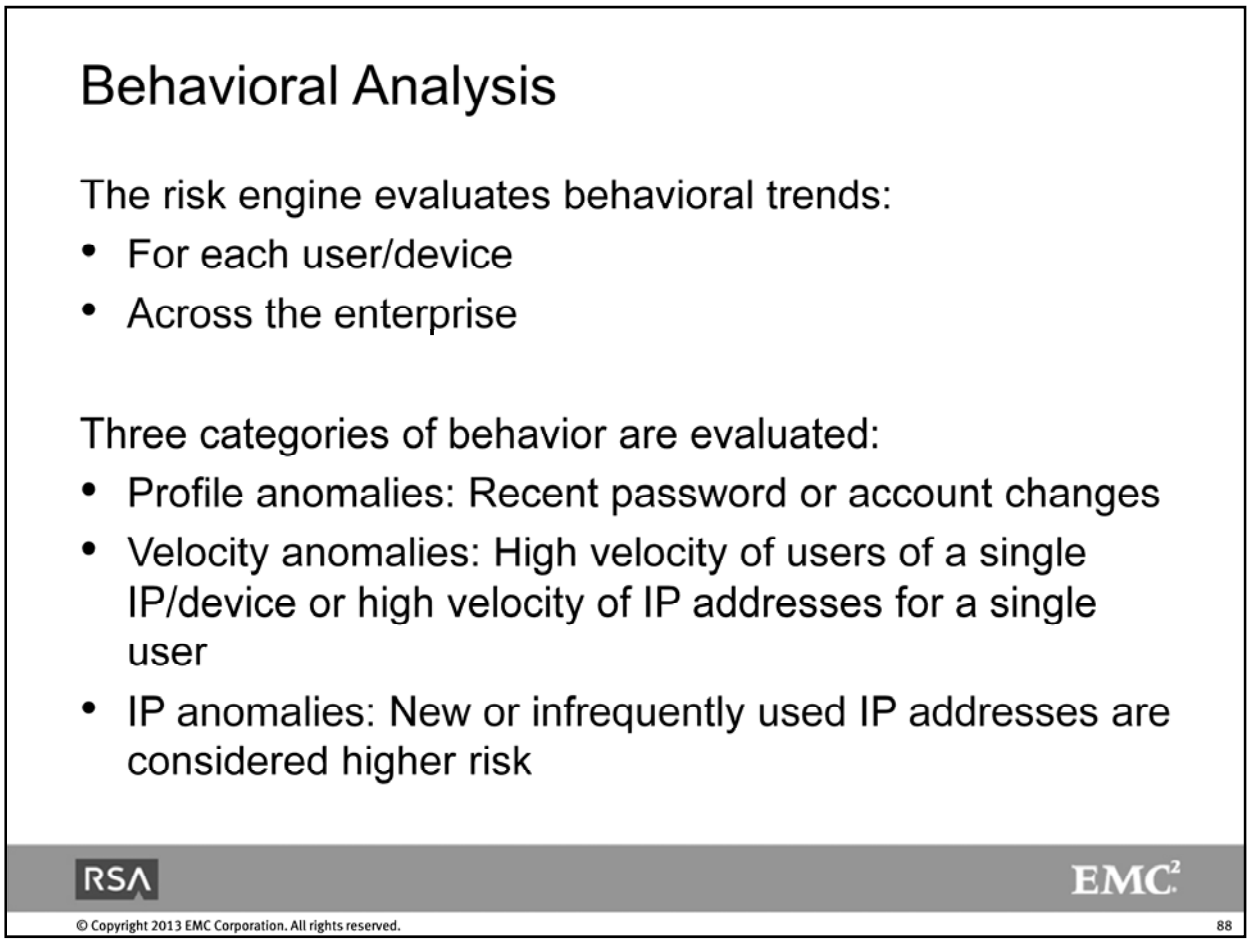
Device tokens are created and placed on the user's machine for future identification using a combination of cookies and Flash Shared Objects (FSOs)

Device Token Recovery can automatically restore user-deleted tokens based on device forensics.

Device Token Theft Protection prevents impersonation of a device using a stolen token (e.g., via malware) through a combination of techniques

Encryption of the device ID in the token prevents reuse on another computer

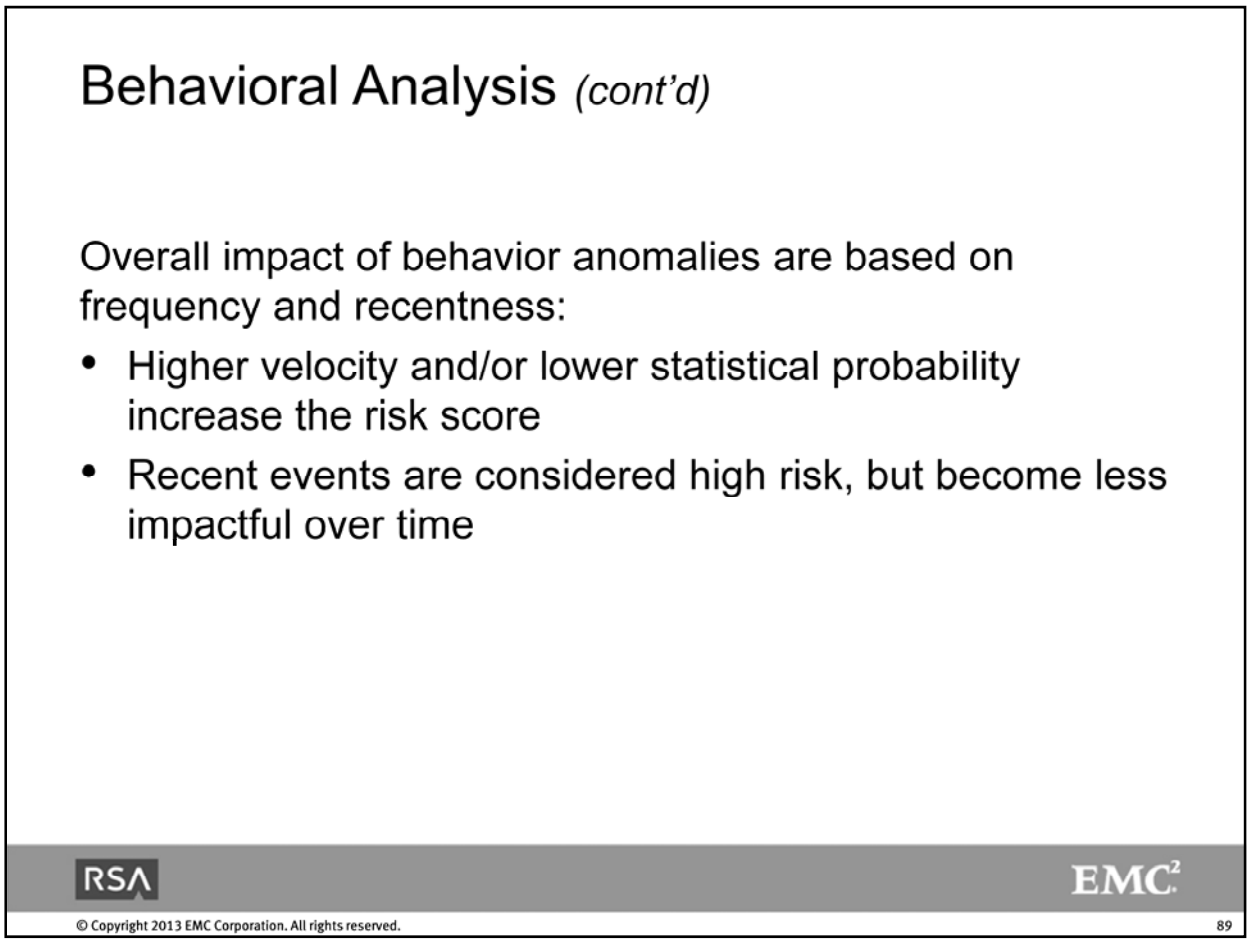
Token generation counter prevents replay of an older token



Behavior is analyzed both for the individual user and across the entire user population. Some behavior, such as abnormal account activity, is always considered suspicious while other behavior, such as authenticating from an unknown location, is suspicious only within the context of the entire user population.

Three categories of behavior are evaluated:

- **Profile anomalies:** Recent password or account changes
- **Velocity anomalies:** High velocity of users of a single IP/device or high velocity of IP addresses for a single user
- **IP anomalies:** New or infrequently used IP addresses are considered higher risk



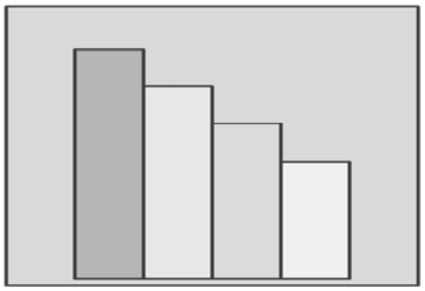
Overall impact of behavior anomalies are based on frequency and recentness:

- Higher velocity and/or lower statistical probability increase the risk score
- Recent events are considered high risk, but become less impactful over time

Minimum Assurance Levels

Four minimum assurance levels are supported:

- High
- Medium-high
- Medium
- Low



RSA EMC²

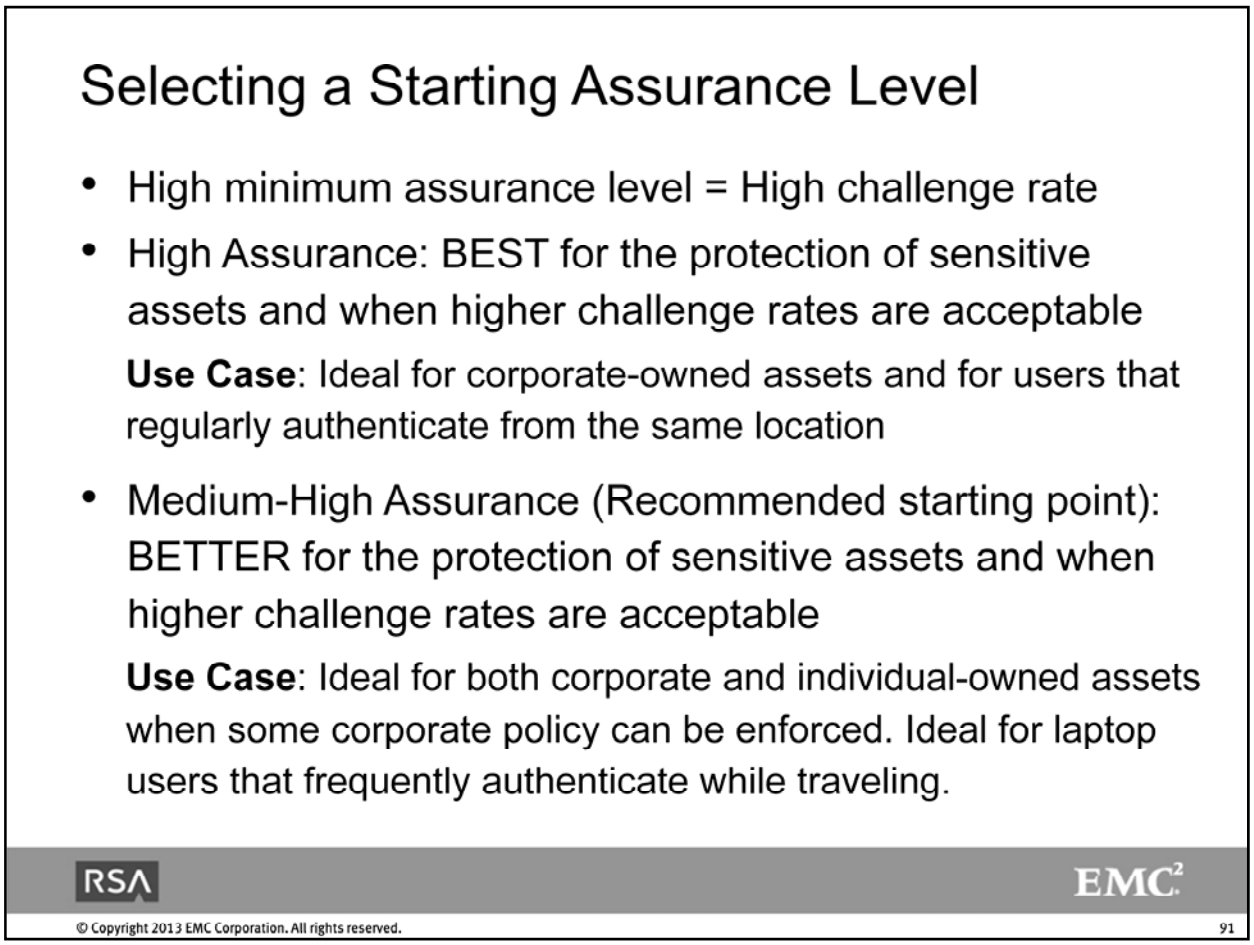
90

Authentication Manager supports four minimum assurance levels: high, medium-high, medium, and low.

The minimum assurance level is a setting that you configure to determine the assurance threshold that each authentication attempt must meet to avoid being challenged for identity confirmation. The setting is in the RBA policy for each user's security domain.

Each time a user authenticates, the risk engine evaluates the device match and user behavior in real-time to produce an assurance level. The risk engine compares the user's assurance level with the minimum assurance level in the RBA policy. If the user's level is lower than the minimum, the user is prompted for identity confirmation.

For example, suppose that the risk engine assesses a user's authentication attempt as medium assurance, and the policy for the user's security domain requires medium-high assurance. The user is challenged for identity confirmation.



Choosing a high minimum assurance level provides stronger protection, but may also result in a rate of users challenged. Because assurance increases when the device is known and user behavior is predictable, some user populations benefit more from a high minimum assurance level than others.

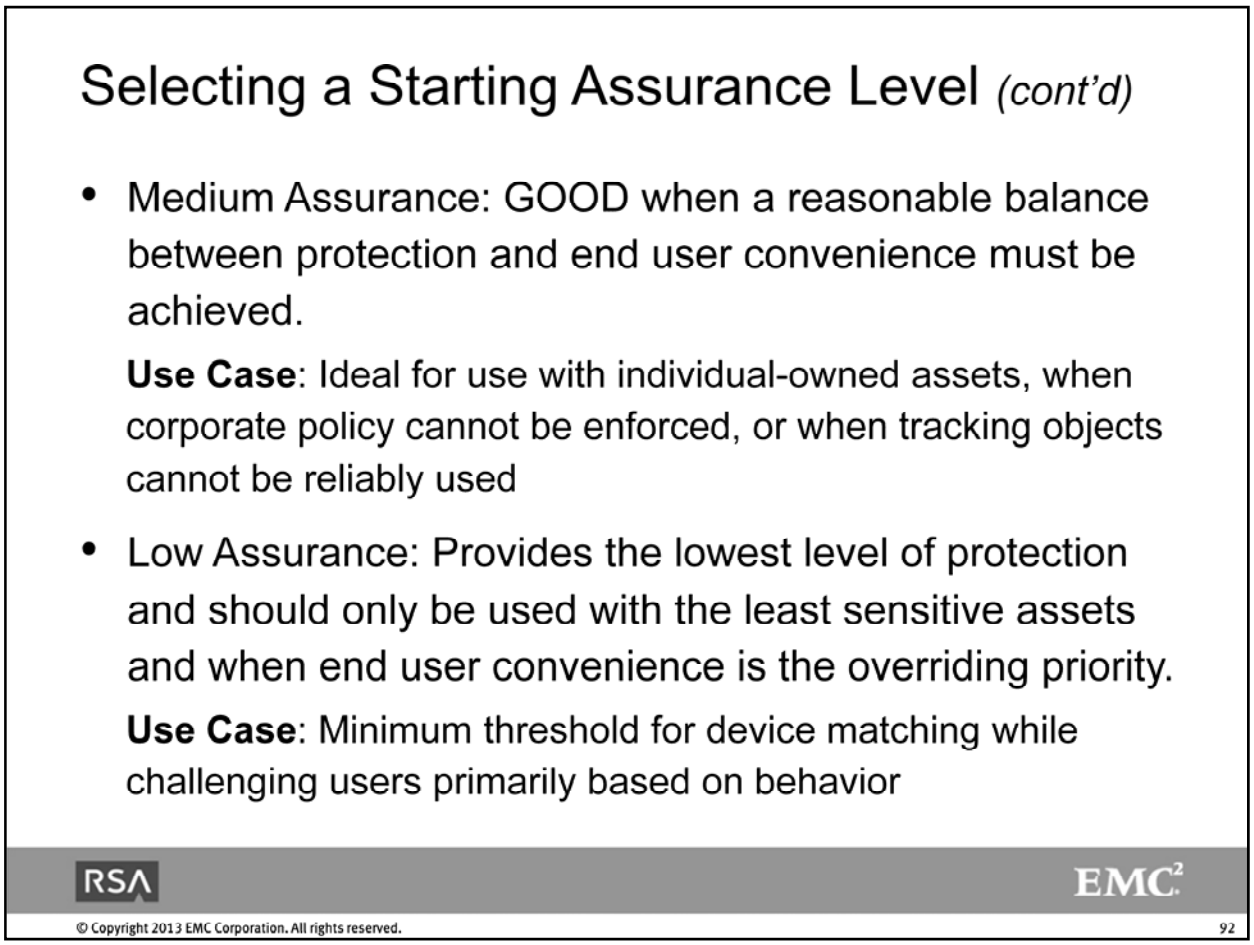
Let's examine the minimum assurance levels.

- High Assurance: BEST for the protection of sensitive assets and when higher challenge rates are acceptable

  Use Case: Ideal for corporate-owned assets (e.g., an employee laptop) and for users that regularly authenticate from the same location (e.g., branch office, partner location, or an employee's home).

- Medium-High Assurance (Recommended): BETTER for the protection of sensitive assets and when higher challenge rates are acceptable

  Use Case: Ideal for both corporate and individual-owned assets when some corporate policy can be enforced (e.g., cookies must be enabled). Ideal for laptop users that frequently authenticate while traveling.



- Medium Assurance: GOOD when a reasonable balance between protection and end user convenience must be achieved.

  Use Case: Ideal for use with individual-owned assets, when corporate policy cannot be enforced, or when tracking objects (e.g., cookies or flash shared objects) cannot be reliably used (e.g., incognito or stealth browsing)

- Low Assurance: Provides the lowest level of protection and should only be used with the least sensitive assets and when end user convenience is the overriding priority.

  Use Case: Minimum threshold for device matching, while challenging users primarily based on behavior

Silent Collection

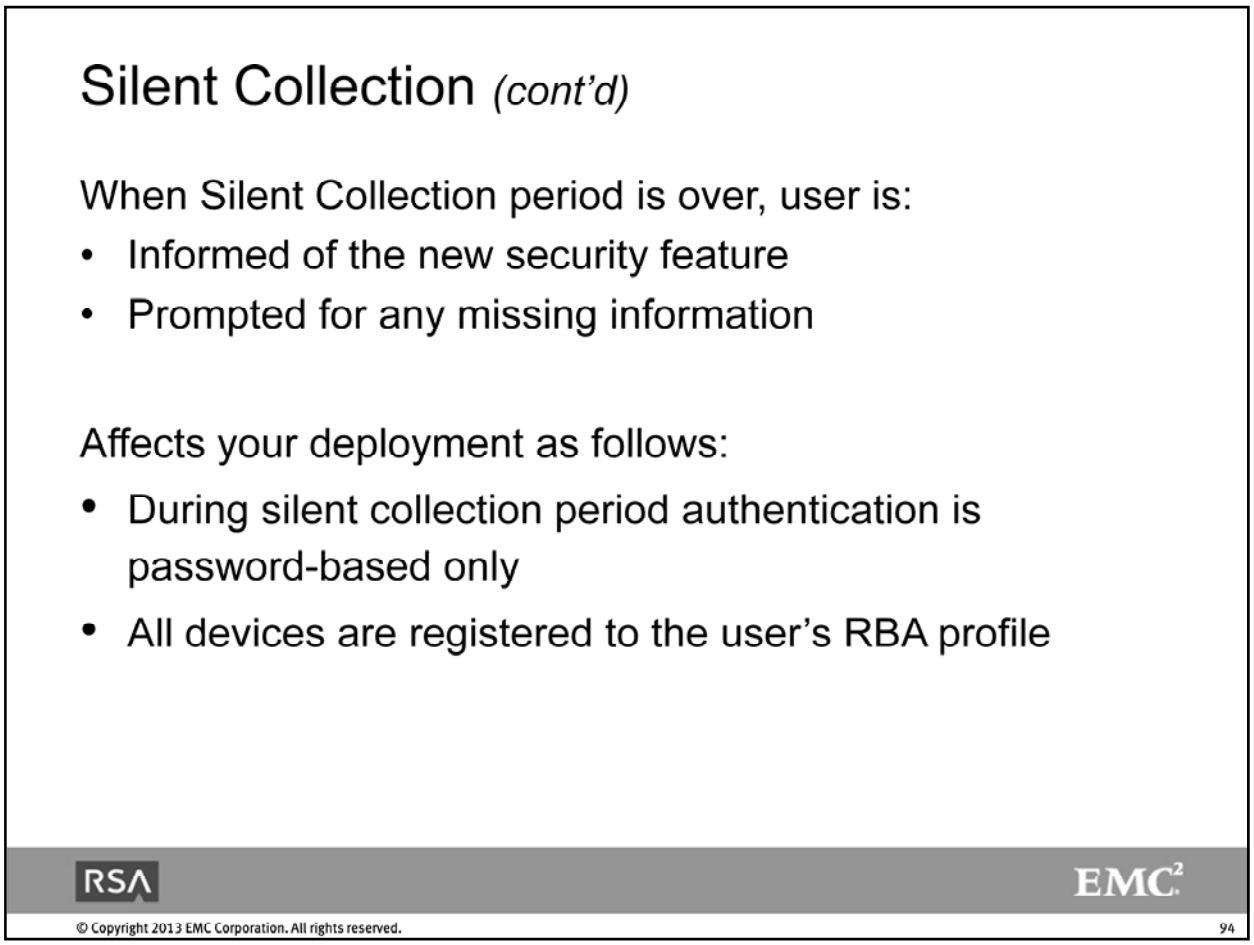
- Simplifies migrating users from password-only to multi-factor authentication with RBA
- Collects profile and behavioral data without administrator intervention
- Is enabled for all users in a security domain
- Helps the risk engine build a baseline profile for each user

RSA EMC²

93

Silent collection is an optional feature that simplifies the process of migrating users from password-only authentication to multi-factor authentication with RBA. Silent collection facilitates the process of collecting profile and behavioral data for new users without the need for administrator intervention.

When silent collection is enabled, the risk engine passively monitors user behavior for a defined period without actively challenging users based on risk. During this period, the risk engine automatically registers user devices and observes behavioral patterns.

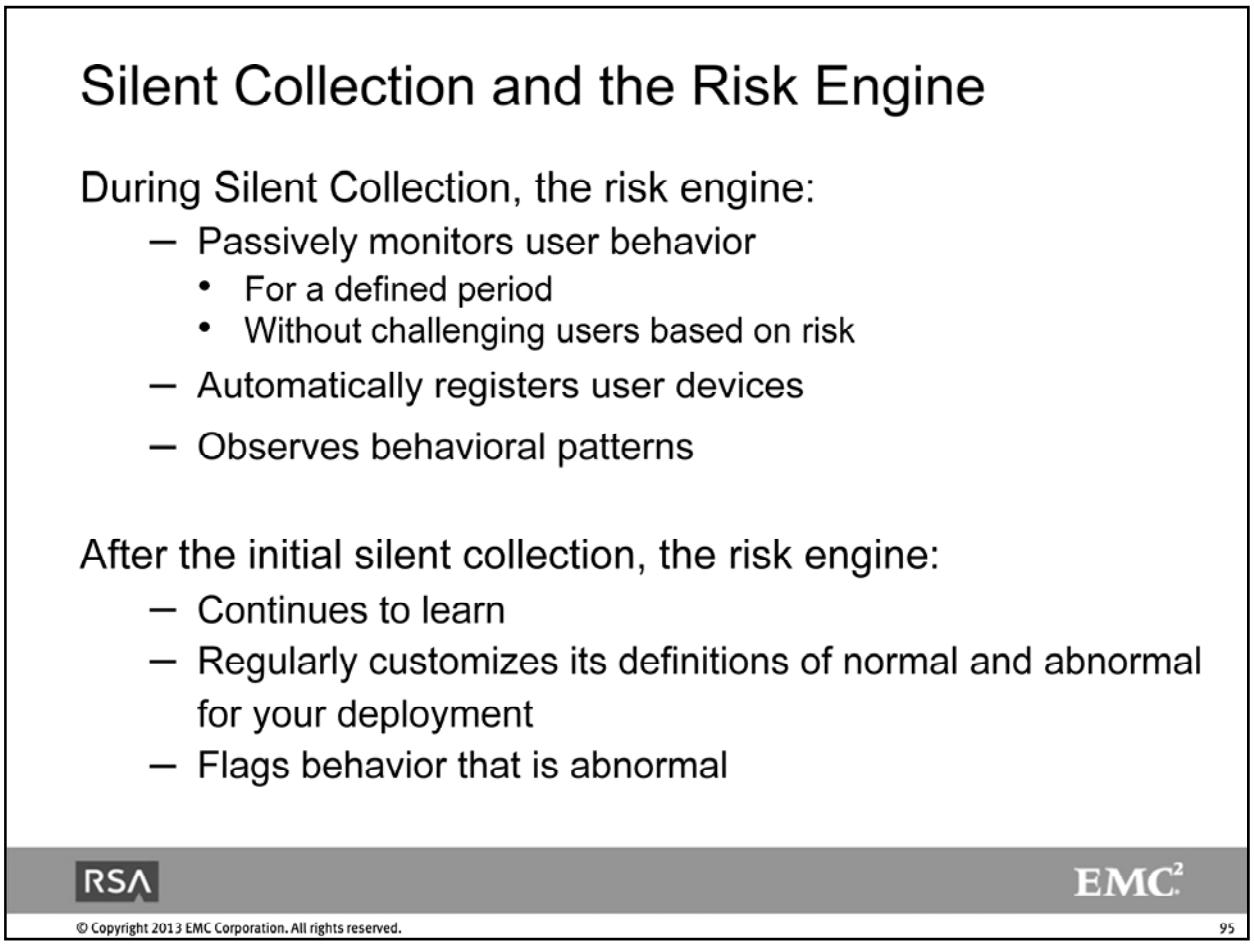


Once the risk engine has gathered enough information to have high assurance about a particular user, that user is informed of the new security feature and is prompted to provide any missing information, such as complete security questions or provide a mobile number.

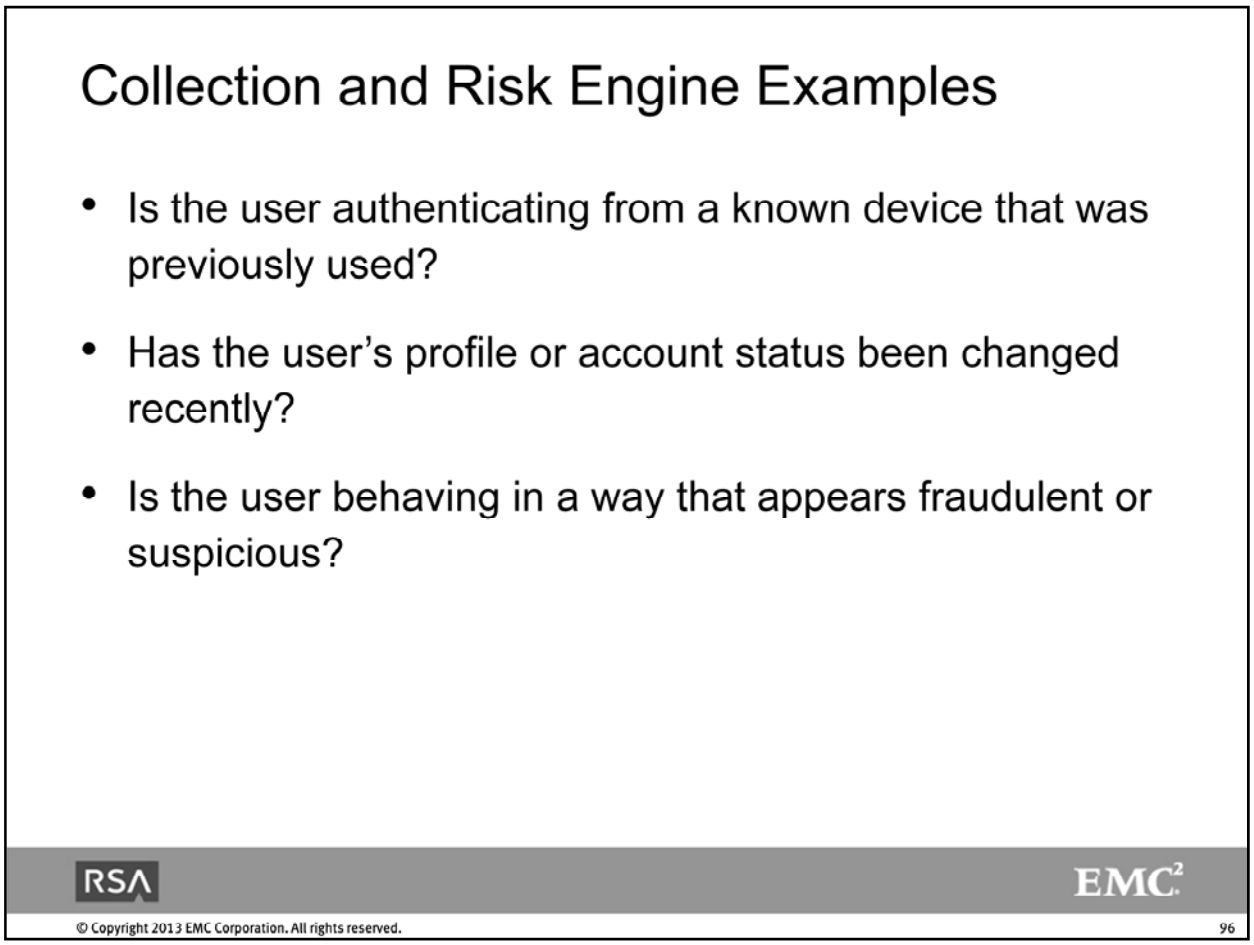
You enable the silent collection period in the RBA policy for all users in a security domain. Using silent collection helps the risk engine build a baseline profile for each user.

Silent collection is optional. It affects your deployment in the following ways:

- During silent collection, authentication is password-based only. Users are never challenged for identity confirmation.
- All devices are registered to the user's RBA profile. This may include unwanted devices such as internet kiosks.



After the silent collection period has expired, the risk engine continues to learn from the behavior of the user population and regularly customizes its risk model to adjust its definitions of normal and abnormal for your deployment. It flags behavior that is abnormal compared with other users.



For example, the risk engine can determine such things as:

•Is the user authenticating from a known device that was previously used?

•Has the user's profile or account status been changed recently?

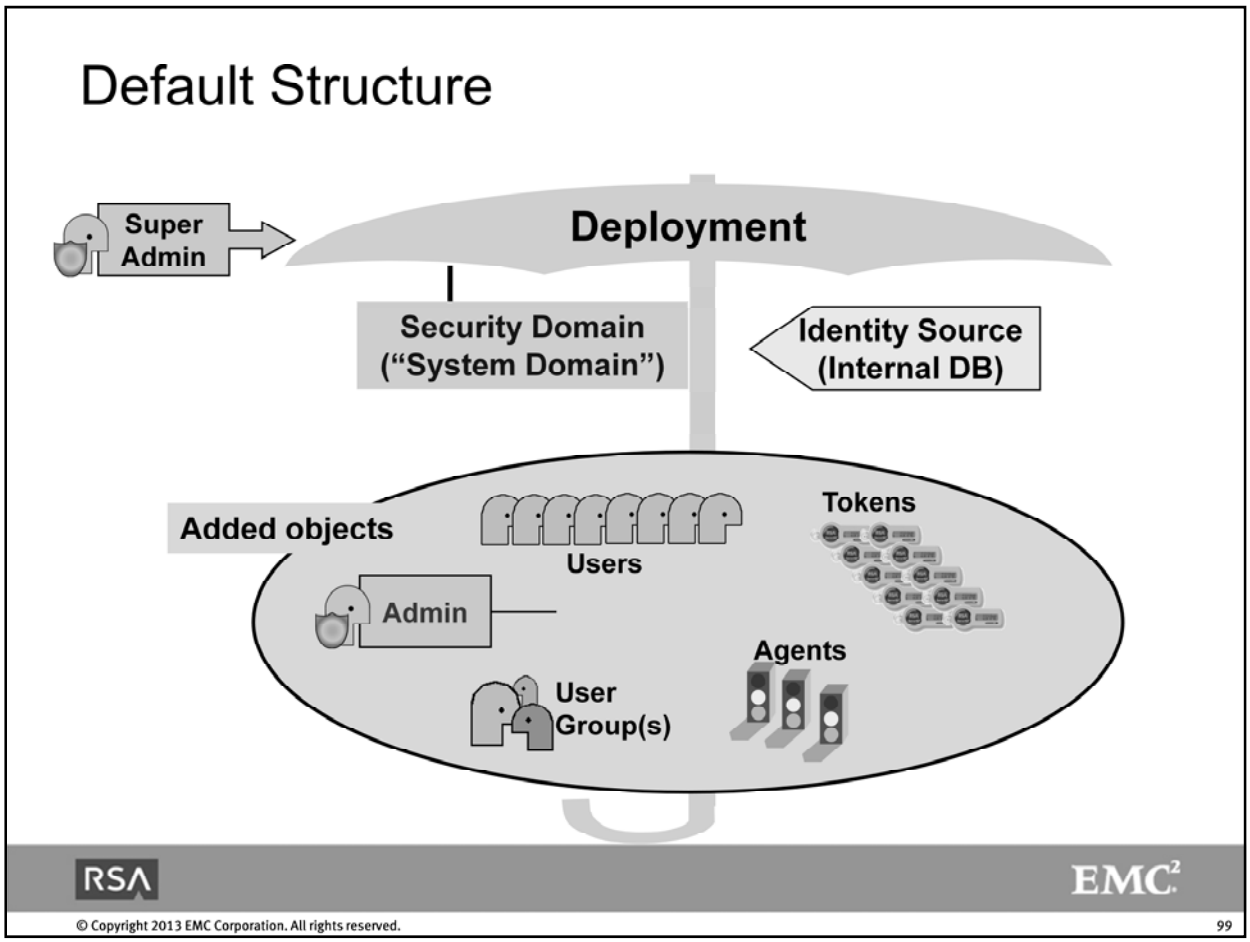
•Is the user behaving in a way that appears fraudulent or suspicious?
For example, is the user attempting to log on from several different devices within ten minutes and failing challenges at the same time?

Unit 3 Quiz

1. What happens during RBA Silent Collection?
2. What are some elements of the Device Fingerprint?
3. What are the authentication factors in the case of RBA?
4. Are “seed records” necessary for On-demand Authentication?
5. If the RBA assurance level is configured as ‘HIGH’, is the relative risk high or low?

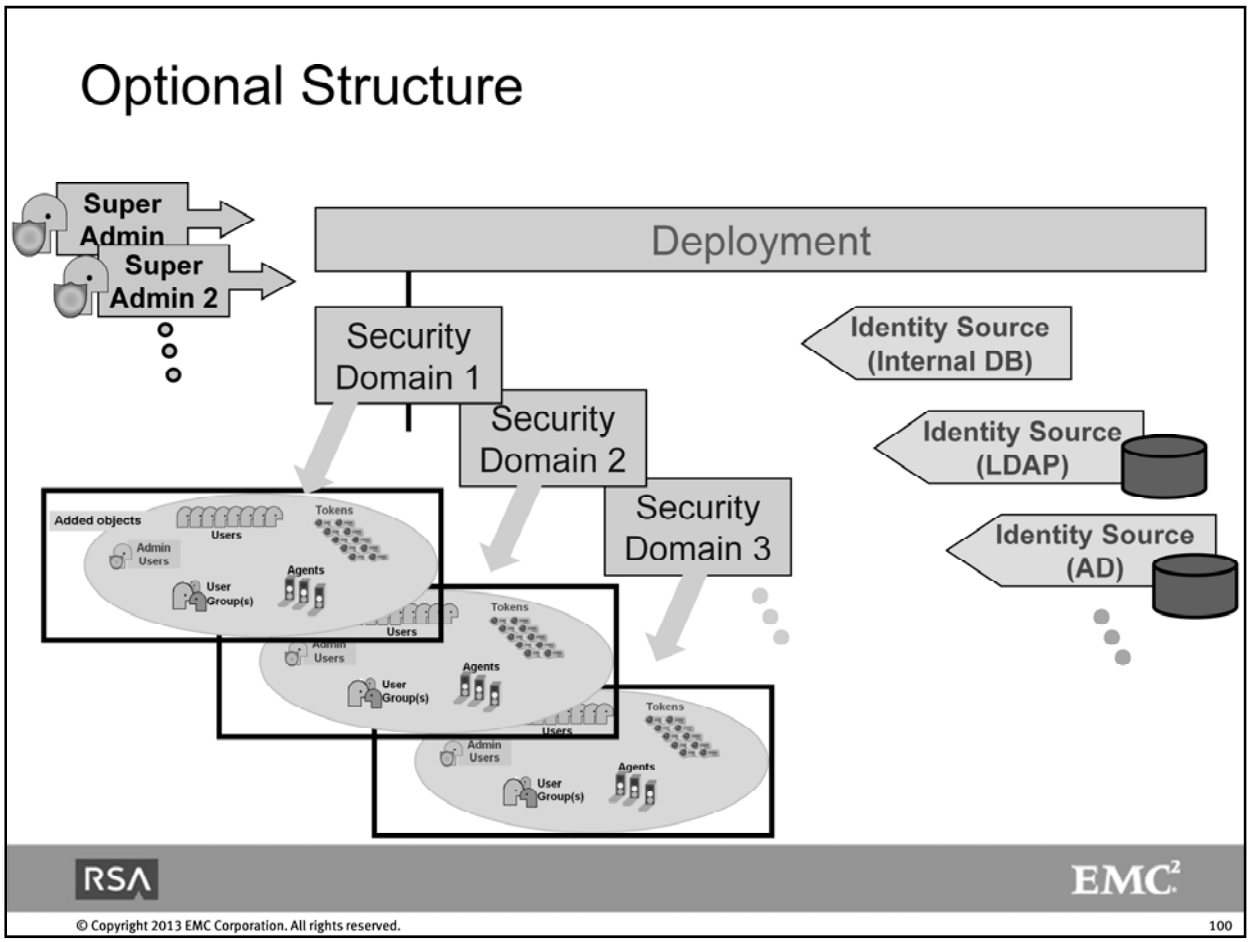
Deployment and Administrative Structure

Unit 4



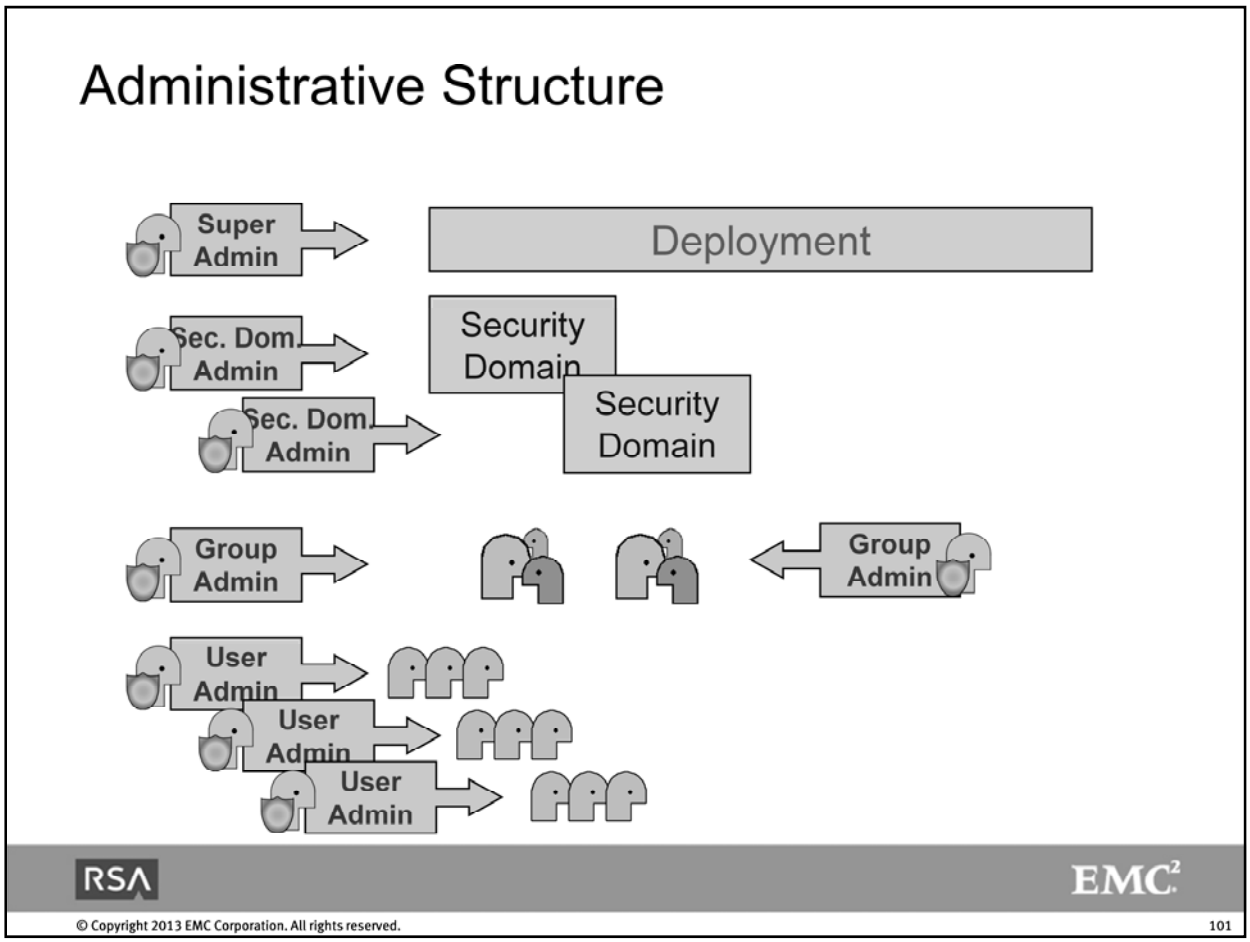
This slide shows the default objects at installation – SystemDomain and Identity Source (internal database) and the default Super Admin.

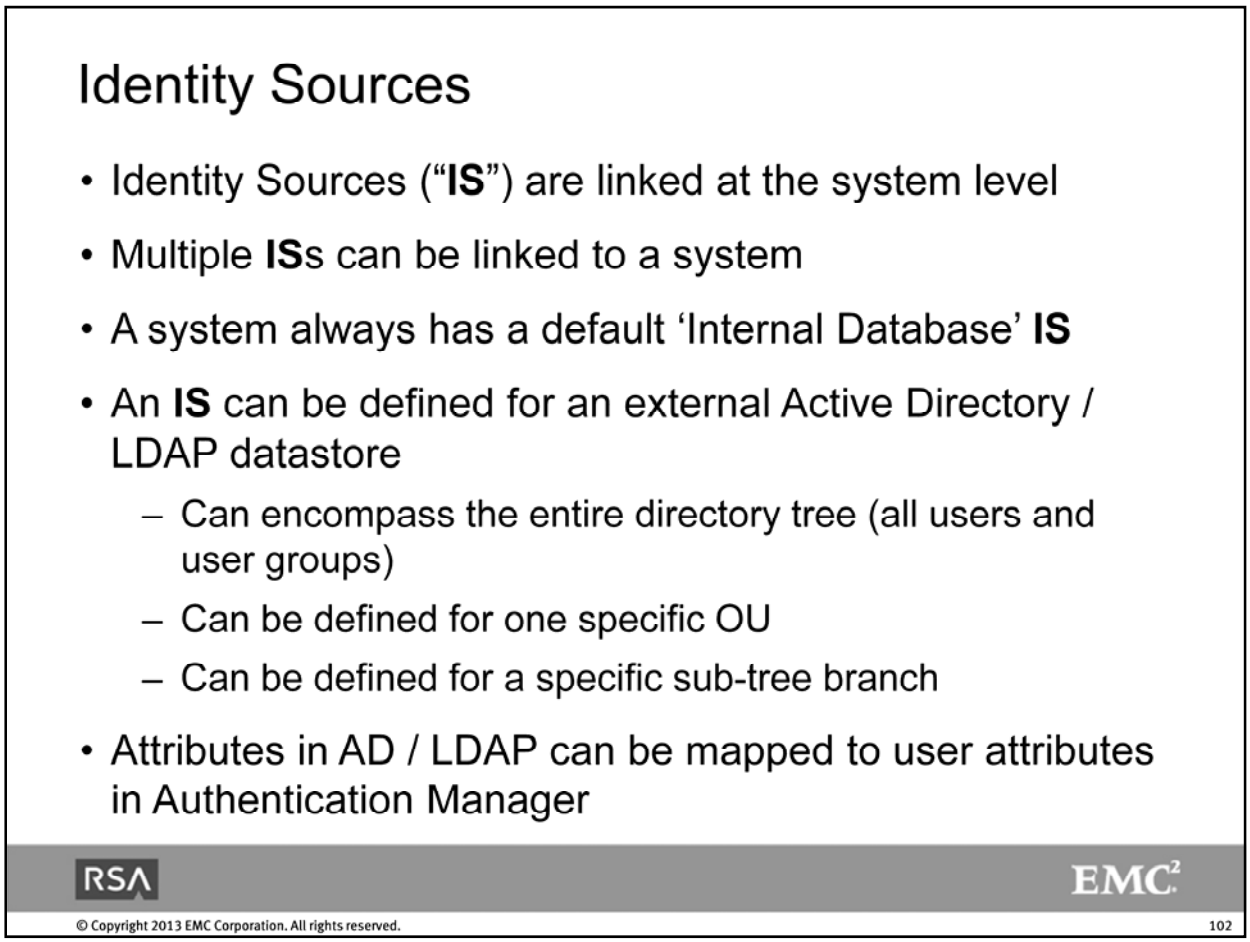
A simple deployment would then add Tokens, users, Agent Hosts and perhaps an additional administrator (at some level – Super Admin or a lower level admin) and perhaps an additional Identity Source.



A more complex structure can be constructed by adding multiple Security Domains, Identity Sources and arranging users into groups.

Additional administrators can be created and given specific scope for the objects – for example, an administrator that can only view and edit members of a certain Security Domain.

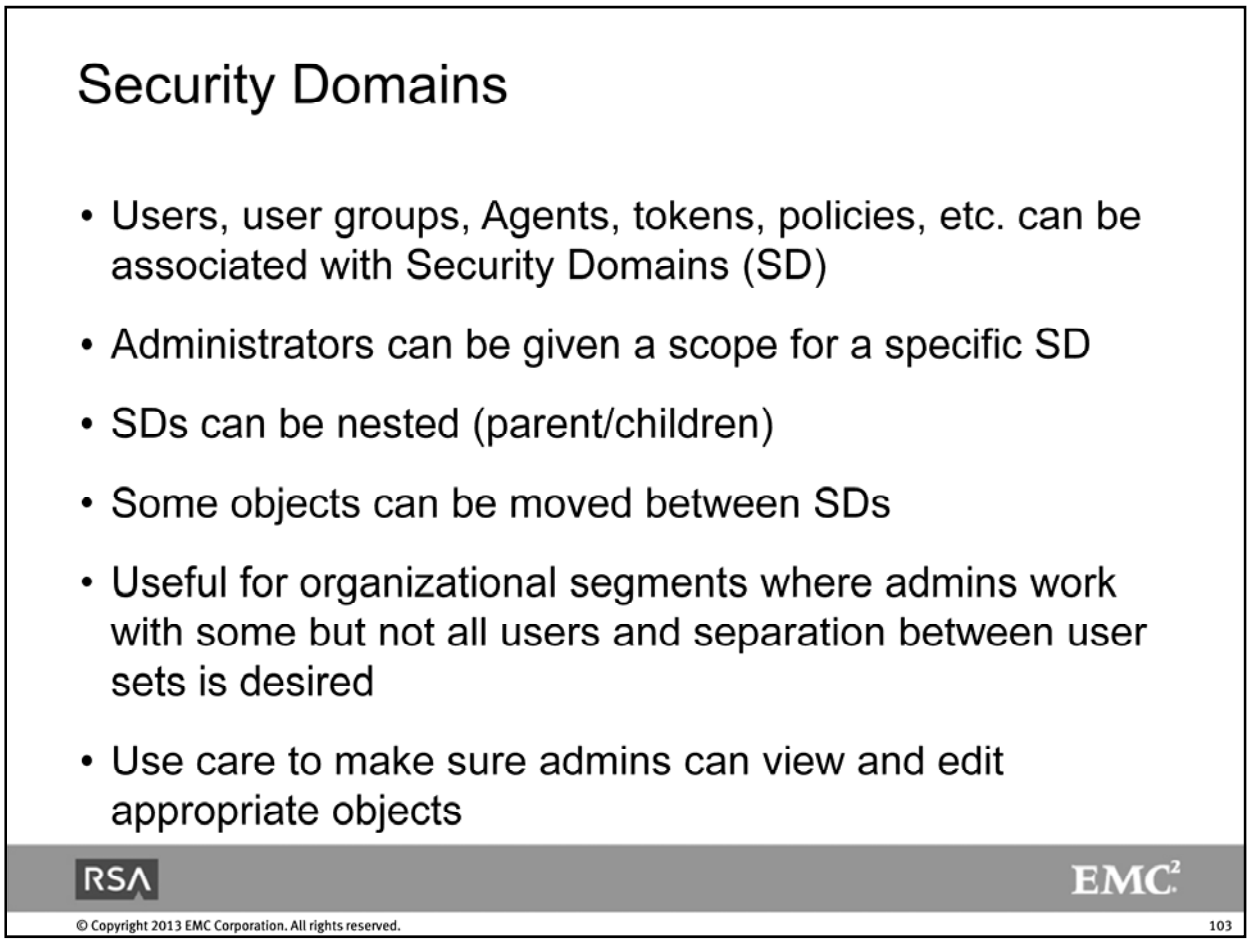

Administrative Structure
Super Admin
Deployment
Sec. Dom. Admin
Security Domain
Sec. Dom. Admin
Security Domain
Group Admin
Group Admin
User Admin
User Admin
User Admin
RSA
EMC²
© Copyright 2013 EMC Corporation. All rights reserved.
101



Identity Sources are linked to Authentication Manager at the system level.

An Identity Source ("IS") can be defined for specific OUs and containers within an Active Directory / LDAP structure. So, for example, one IS can be defined for all users within a 'Sales' OU, another IS can be defined for users within an 'IT Dept' OU and so forth.

User attributes can be mapped from an Identity Source to Authentication Manager as appropriate. For example, an employee telephone number existing in LDAP might be useful for an Authentication Manager administrator to have as part of the user account if support or other issues require the administrator to call the user – or a mobile phone number can be registered for a user to receive On-Demand Authentication codes.



Security domains represent areas of administrative responsibility. All Authentication Manager objects other than Identity Sources are managed within a security domain. Security domains allow you to:

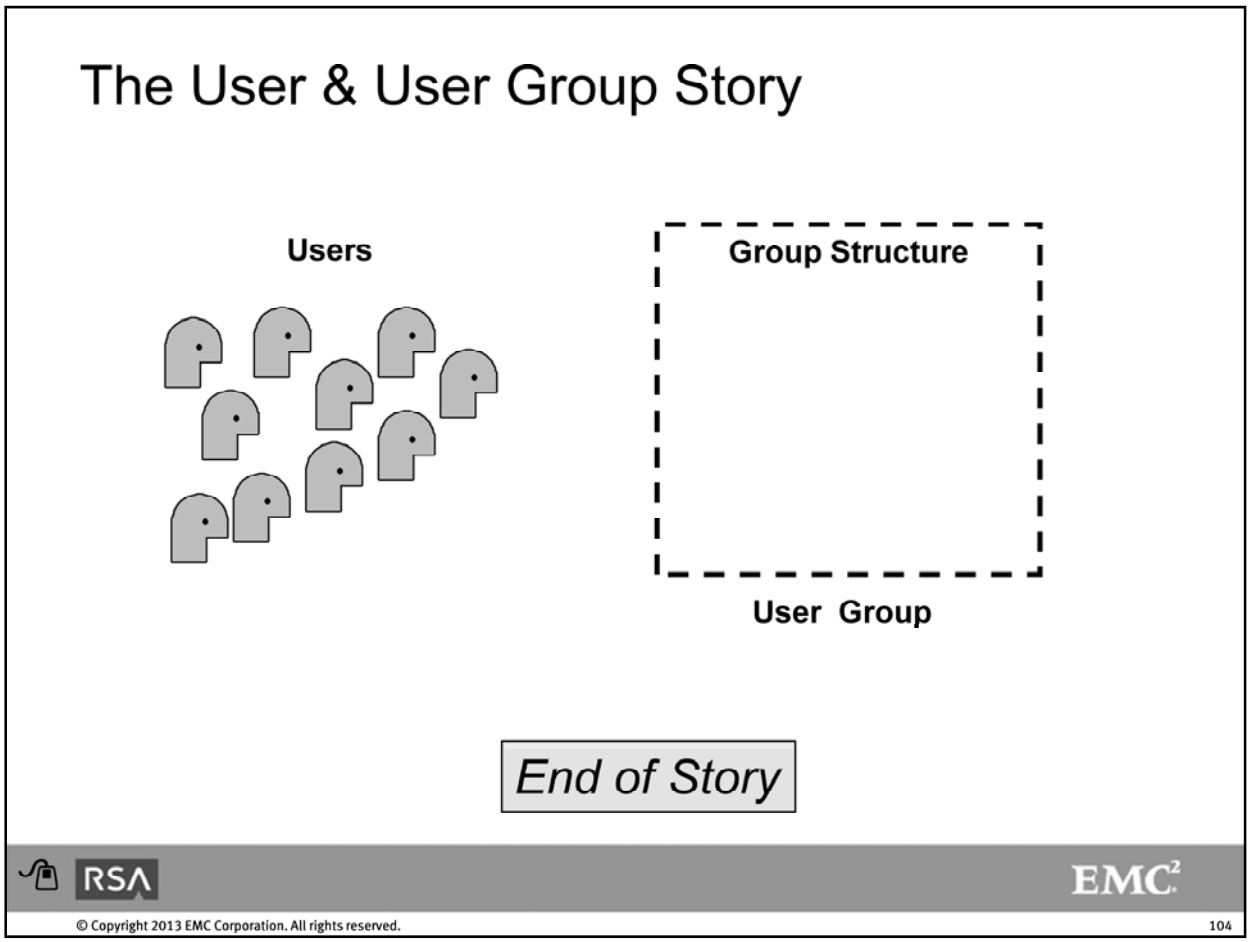
- Organize and manage users
- Enforce system policies
- Limit the scope of administrators' control by limiting the security domains to which they have access.

By default, all users are managed in the top-level security domain. You can transfer users from the top-level security domain to other security domains. For example, you can create separate security domains for each department, such as Finance, Engineering, and Human Resources, and then move users and user groups from each department into the corresponding security domain. Security domains are often created to mirror the departmental structure or the geographic locations of an organization.

Security Domains can be nested such that a domain becomes a child to an upper-level “parent” domain (security domains ‘East’ and ‘West’ under a ‘Sales’ domain, for example).

Security domains can also be used to enforce system policies.

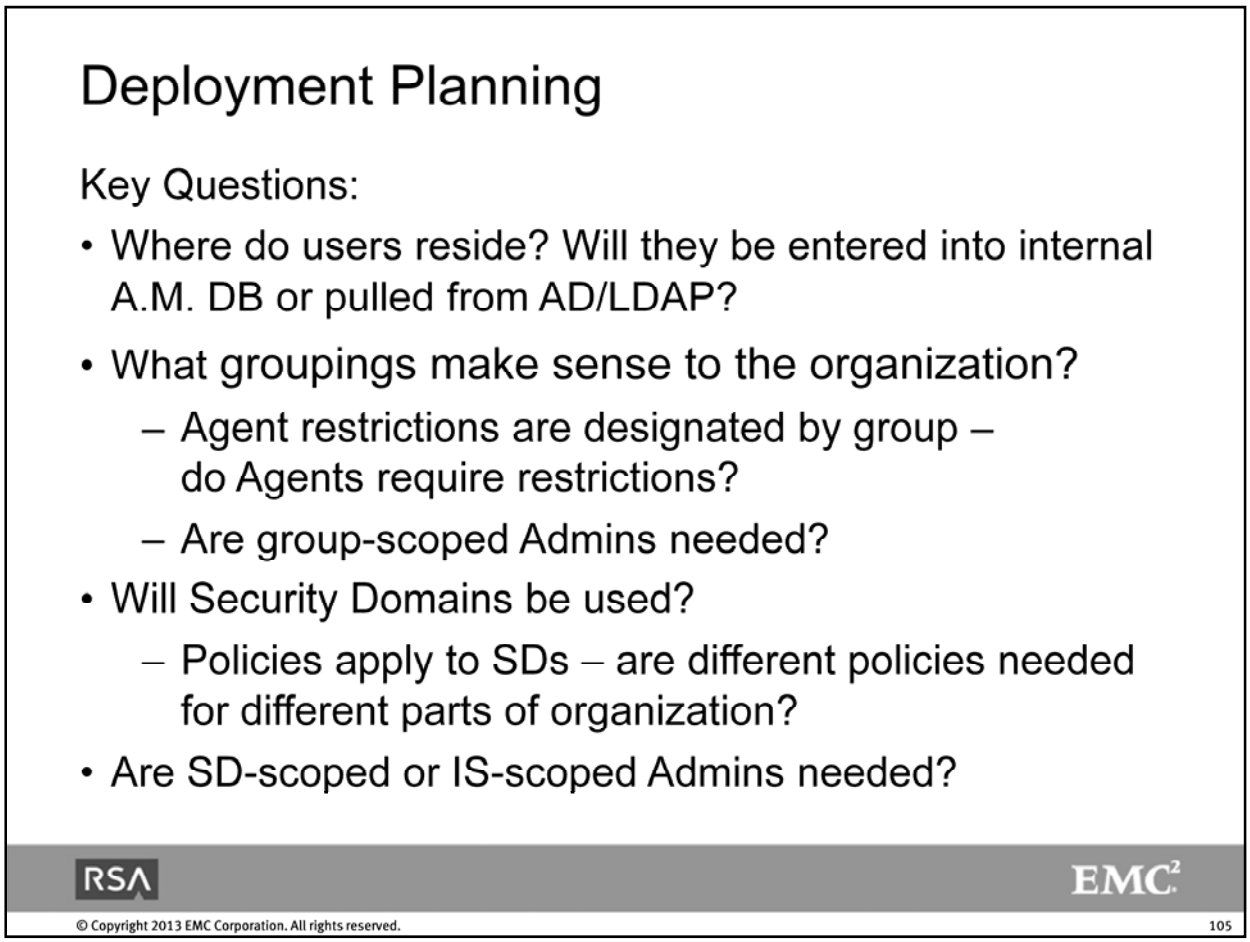
Note that the policy assigned to a lower-level security domain is not inherited from upper-level security domains. New security domains are assigned the default policy regardless of which policy is assigned to ‘parent’ security domains above them in the hierarchy



Group structures are useful for general organizational purposes - for example, having all users in the Sales Department as members of a 'Sales' group. This might benefit reporting or managing token deployments.

Groups can also be nested within other groups - for example, a group 'Sales' containing sub-groups of 'East' and 'West' or 'Internal' and 'Field Sales'. In the latter case, the use of a 'Field Sales' group would be useful if a new VPN server was installed and you wished to activate all external sales users on that Agent at one time.

If an organization is large enough to require many administrators, establishing User Groups can support administrative responsibilities. For example, an administrator located in an a manufacturing facility might be assigned a role that allowed editing only users in the 'Manufacturing' group but not users in any other groups. This allows separation of administrative functions at a very granular level.



In the set of user documentation provided with RSA Authentication Manager, there is a comprehensive Planning Guide that helps you through the process of devising a strategy to handle these and other issues.

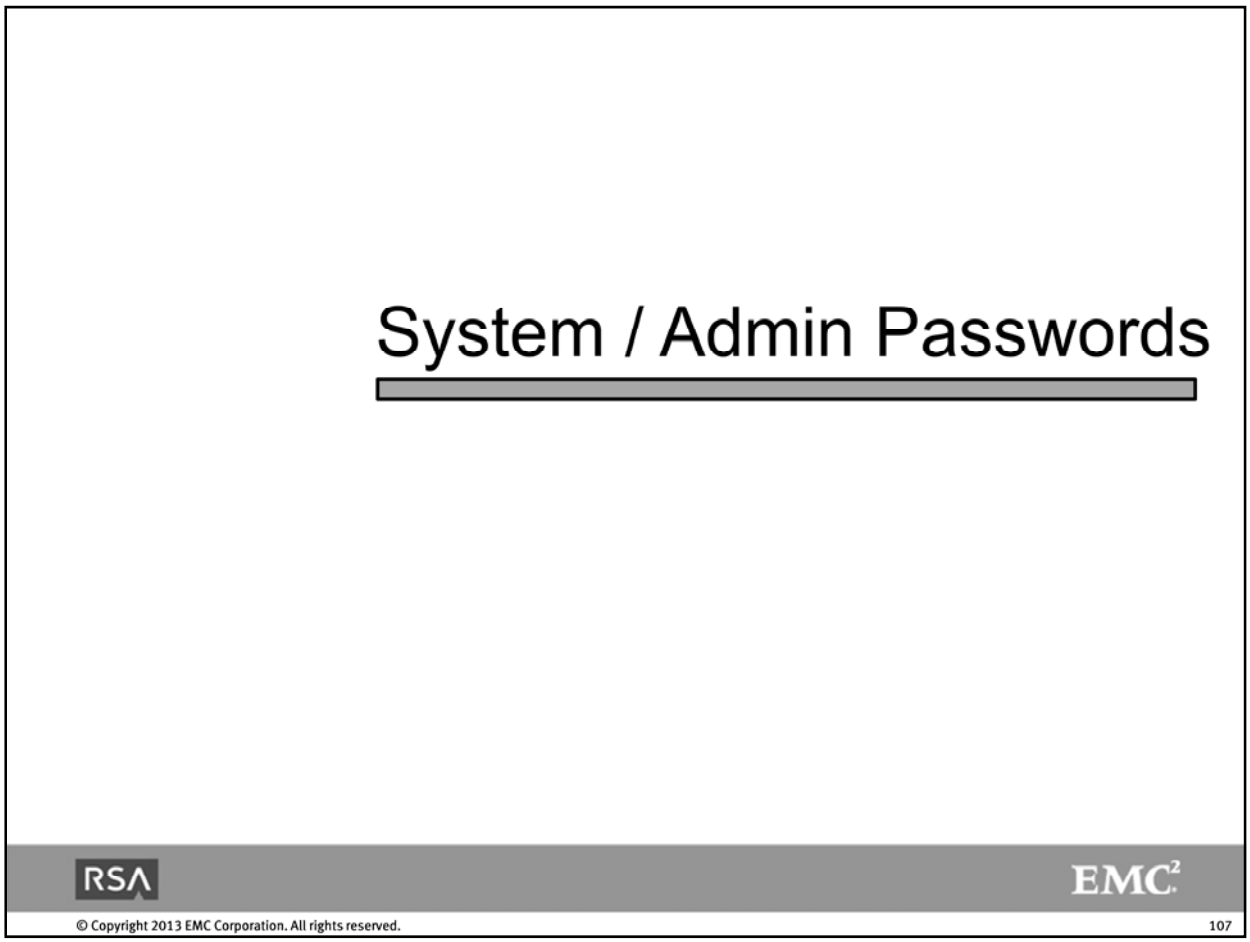
Deployment Planning *(cont'd)*

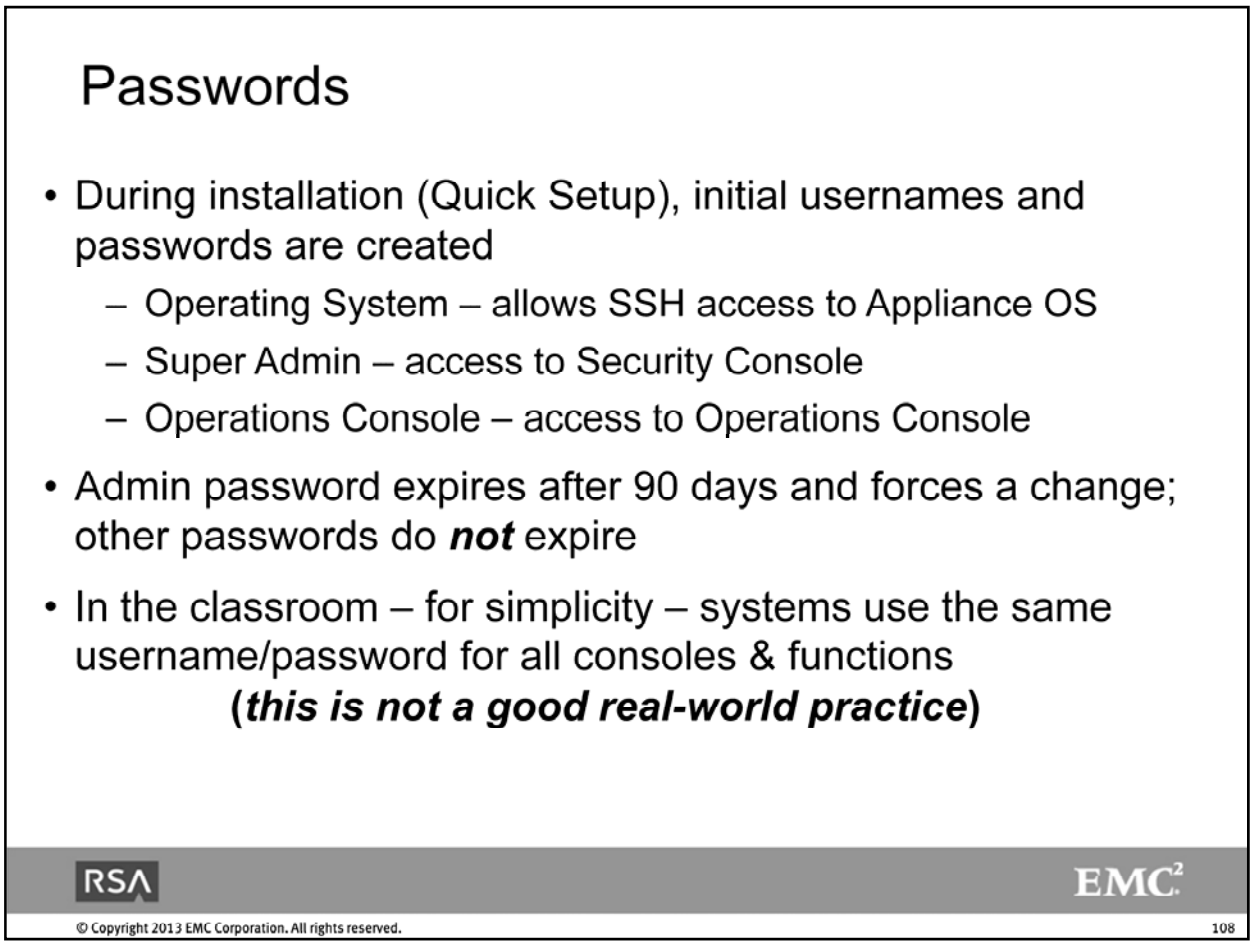
- Number and type of Administrators
  - Varying tasks and scopes (HelpDesk, Users, Tokens, Reporting, etc.)
- Polices for authentication, lockout, offline and emergency access
- Token issue and deployment
  - Secure delivery; Options for self-service and provisioning
- Authentication methods
  - Differing user types and user groups may require different methods

User documents include a comprehensive Planning Guide that discusses these issues.

RSA EMC²

106


System / Admin Passwords
RSA
EMC²
© Copyright 2013 EMC Corporation. All rights reserved.
107



During installation, usernames and passwords are established for:

- The initial Super Admin administrator
- The Operations Console administrator

The initial 'Super Admin' password expires after an initial 90 day period and after that period the Super Admin administrator will be prompted to change it - if it has not already been changed.

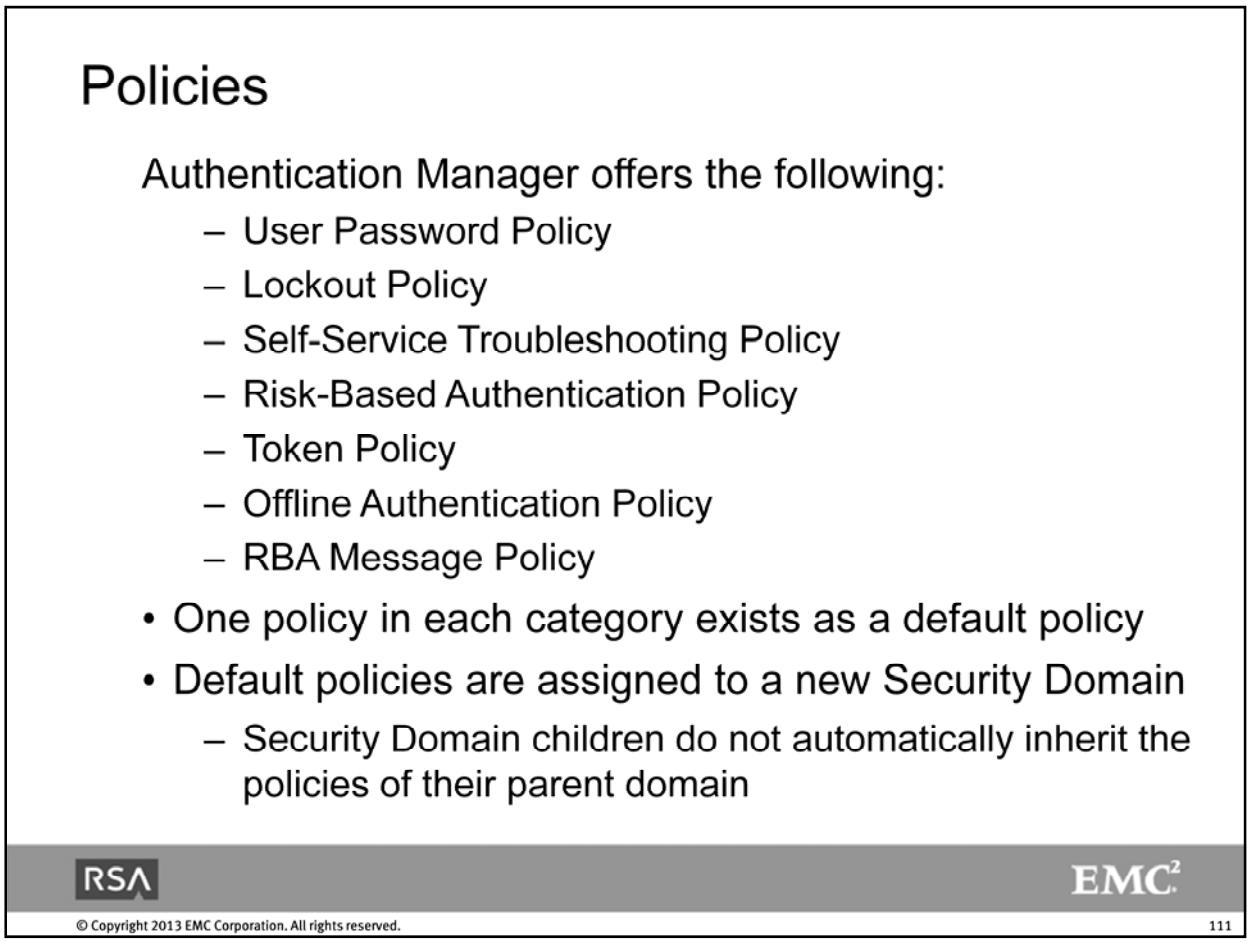
If security policy permits, it may be advantageous to record the key account password information as part of the system documentation and stored in a secure location.

Unit 4 Quiz

1. What structural element can have specific policies associated with it?
2. How wide is the time window that the server allows for a time-synchronous tokencode?
3. What Identity Source is *always* a part of the Authentication Manager Structure?
4. Administrative scope can be limited to what part of a deployment's structure?
5. What administrative password automatically expires 90 days after Authentication Manager installation?

Policy Management

Unit 5

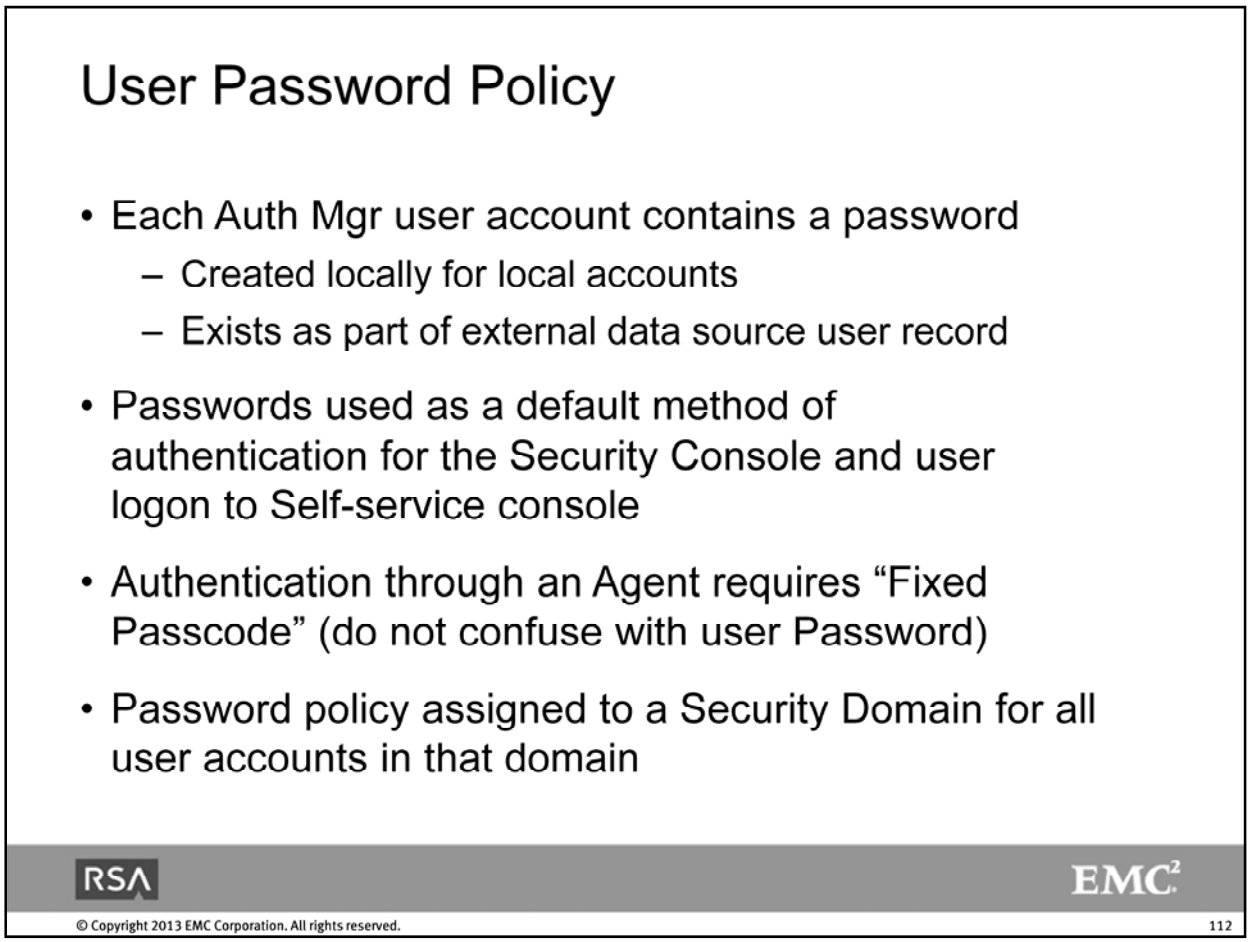


RSA Authentication Manager allows you to create a number of authentication policies and assign those polices to Security Domains.

You can create a number of different policies for various applications and groups of users. One policy is assigned to a Security Domain (users within one given domain can not be assigned different policies).

From your set of policies, one policy is designated as the default policy for the system. Each new Security Domain created will initially use that default policy. You can designate a different policy at the time that you create a security domain or by editing a security domain.

A policy assigned to an upper-level Security Domain is not inherited by Security Domains created beneath it (domain children). A new Security Domain, even if created as a child of another domain will initially use the default policies.



All RSA Authentication Manager users are required to have a password as part of their user account record. If the Authentication Manger internal database is used as the user identity source, the password is stored in the internal database.

Password characteristics are controlled by the user Password Policy, which define users' password length, format, and frequency of change.

Do not confuse Authentication Manager user Passwords with "Fixed Passcodes". Although similar in nature, Fixed Passcodes are used when a user authenticates through an Agent and is used like a token Passcode - user Passwords are used only for Console access in Authentication Manager.

The user password is used by administrators, if policy permits, to log on to the RSA Security Console or for users logging on to the Self-service Console.

If an external LDAP directory is used as the identity source, the password field in the Authentication Manager user record may be mapped to the LDAP password. Password characteristics for users created in an LDAP directory are dictated by the applicable LDAP polices. Authentication Manager password policies apply only to LDAP users when the user is edited from within the Authentication Manager Security Console.

Password policies are applied to security domains. The password policy assigned to a security domain dictates the password-related requirements for all the users assigned to that security domain.

User Password Policy *(cont'd)*

Password Policy controls:

- Use of system-generated passwords
- Periodic expiration
  - Minimum/Maximum lifetime
- Minimum/Maximum length
- Excluded characters
- Excluded words dictionary
- Character requirements
  - Number & type of characters
  - Alphabetic, numeric, upper/lower case, special

RSA EMC²

113

Password Dictionary

Password Dictionary contains character strings that can not be used as user Passwords or PINs (same dictionary used for both)

- Only one dictionary can be defined for a deployment
- Can contain any number of entries
- Simple text file

RSA EMC²

Exercise: Password Dictionary

- Create a dictionary of excluded words

RSA EMC²

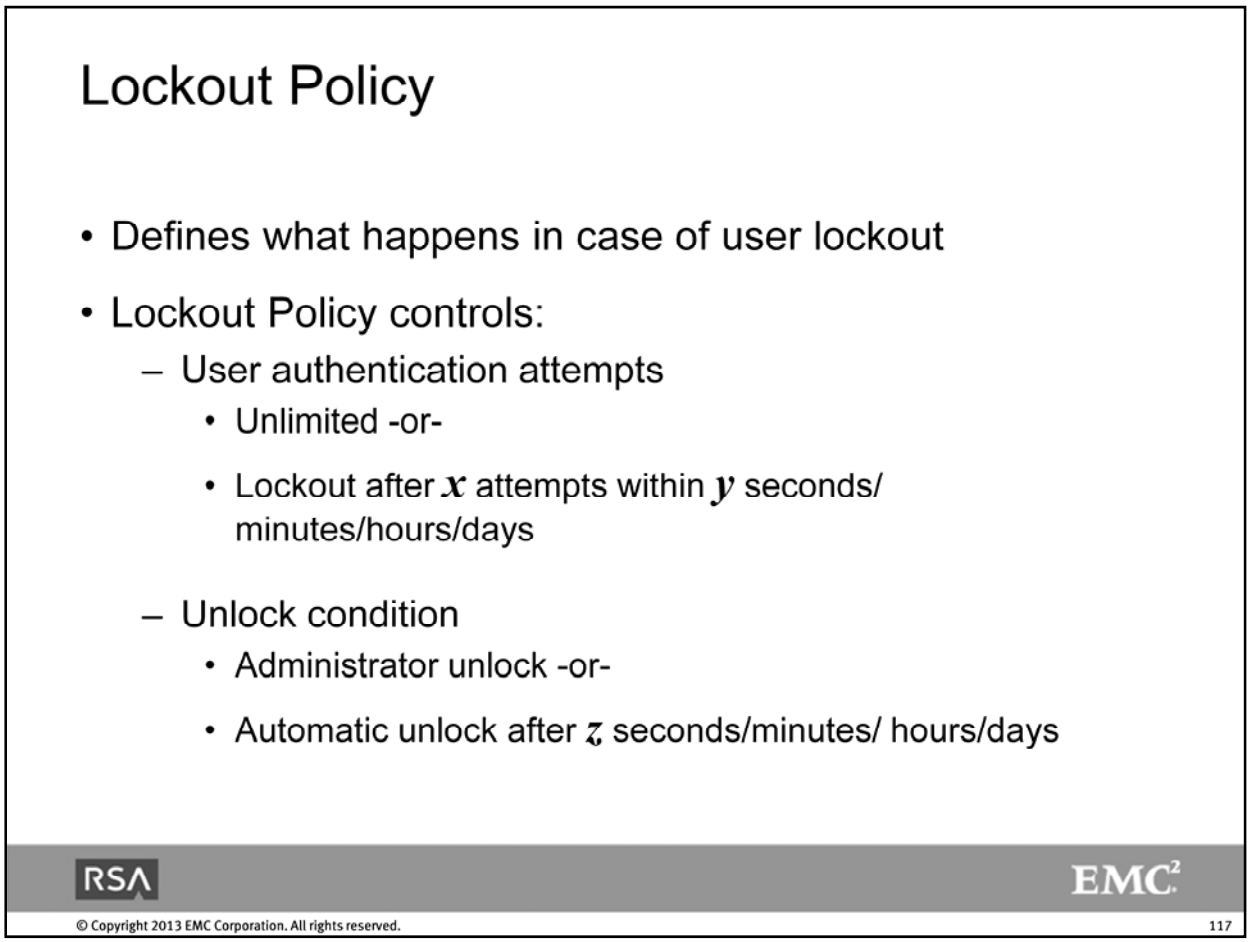
115

Exercise: Password Policy

- Establish policies for both strict and less restrictive passwords

RSA EMC²

116



Lockout policies define how many failed logon attempts users can make before the Authentication Manager locks their account. The Lockout Policy also defines what actions are necessary to re-activate the user after a lockout.

Lockout policies are assigned to security domains. The lockout policy assigned to a security domain dictates the lockout requirements for all the users assigned to that domain.

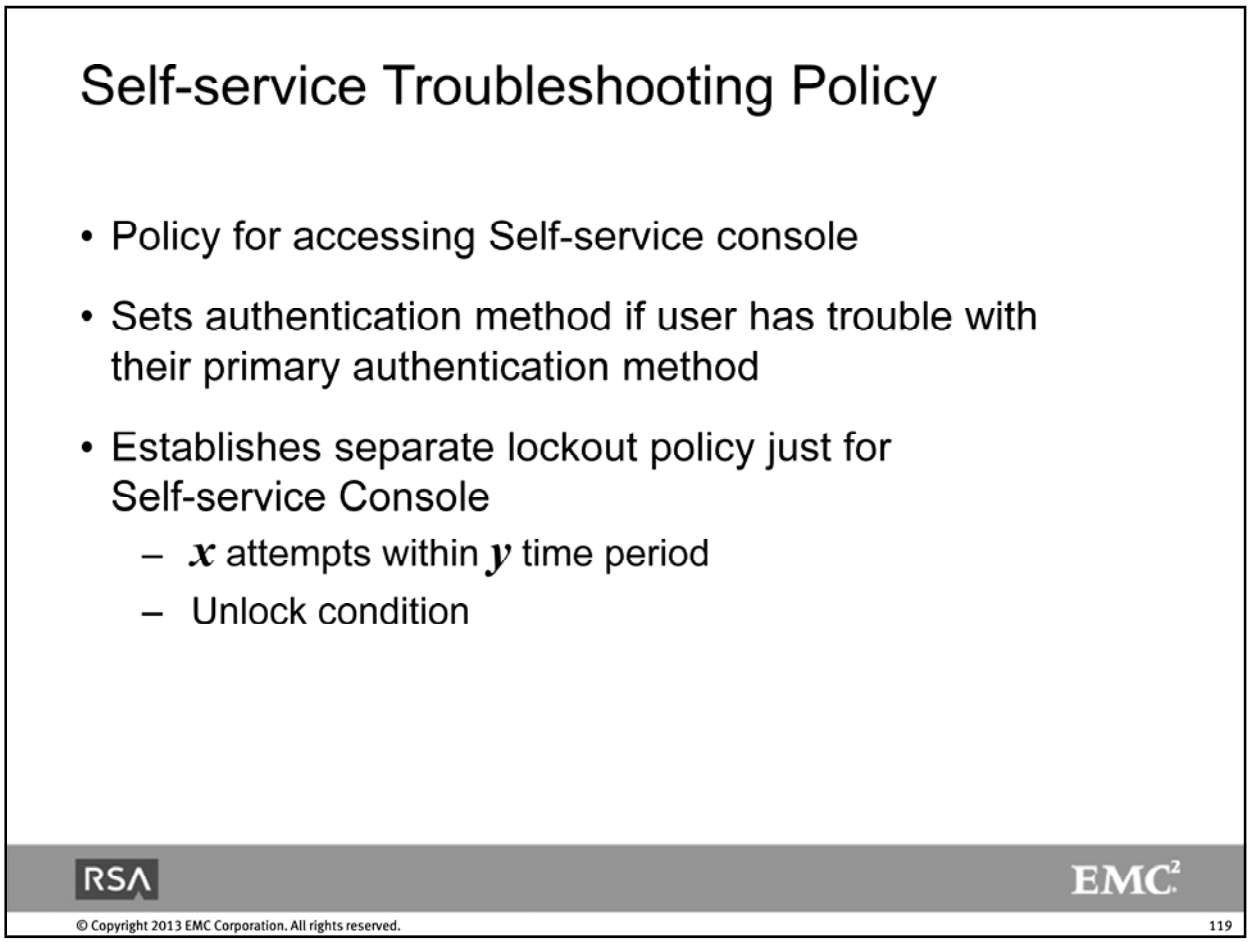
Lockout policies apply to all logon attempts regardless of how many different tokens a user attempts to authenticate with. For example, if a user has two failures with their software token and one failure with their hardware token, that adds up to three failed attempts.

Exercise: Lockout Policy

- Establish lockout policies for users

RSA EMC²

118



Self-service troubleshooting policies apply to Self-Service console users only. These policies allow you to define secondary authentication methods (the default authentication method for access to the Self-service Console is a user Password/LDAP Password).

A secondary authentication method allows a user to access the Self-Service console even if their primary authentication method has failed. You can also use the self-service troubleshooting policy to specify lockout and unlock settings for the Self-service Console.

Similar to the other policies, self-service troubleshooting policies are assigned to security domains. The self-service troubleshooting policy assigned to a security domain dictates the self-service troubleshooting requirements for all the users assigned to that security domain.

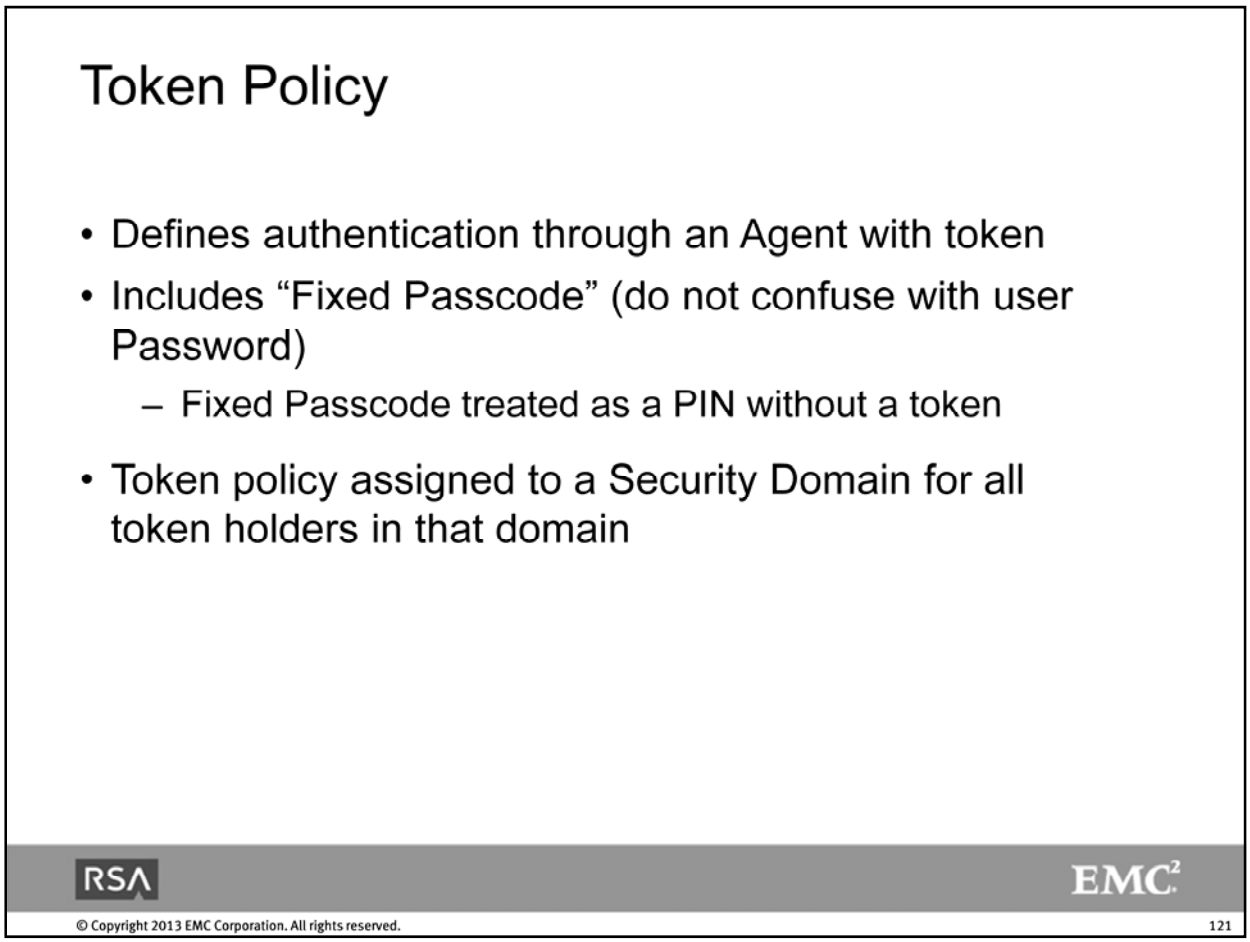
Self-service troubleshooting policies apply to all logon attempts regardless of how many different tokens or authentication methods a user attempts to authenticate with. For example, if a user has two failures with their SecurID token and one failure with their user Password, that adds up to three failed attempts.

Exercise: Self-Service Policy

- Establish policy for Self-Service access

RSA EMC²

120



Token policies define users' RSA SecurID PIN lifetime and format, and fixed passcode lifetime and format, as well as how the system handles users or unauthorized people who enter a series of incorrect passcodes.

Token policies are assigned to security domains. The token policy assigned to a security domain dictates the token-related requirements for all the users assigned to that domain.

When a user authenticates with a token, PIN or fixed passcode requirements are dictated by the token policy for the users' security domain, not by the policy of the token's security domain. For example, if a user assigned to the London security domain authenticates with a token assigned to the Boston security domain, the token policy of the London security domain dictates policy requirements.

When an existing token policy is edited, existing PINs and fixed passcodes are not validated (that is, re-validated) against the excluded words dictionary and history requirements. They are, however, validated against all other policy requirements.

Token Policy *(cont'd)*

Token Policy controls:

- Handling incorrect passcode entries
  - Allow unlimited attempts
  - Put user into Next Tokencode Mode after ***x*** attempts
- Event-based Token range
  - Use default from token records
  - Limit for one code acceptance (default = 3)
  - Limit for Next Code acceptance (default = 7)

RSA EMC²

Token Policy *(cont'd)*

- Require periodic PIN expiration (and min/max lifetime)
- Restrict PIN reuse
  - Cannot reuse ***x*** PINs
  - Can reuse any PIN
- PIN Format
  - User-generated or System-generated
  - Min/Max length (4-8)
  - Excluded words dictionary
  - Character requirements
    - alpha, numeric, alphanumeric
    - require ***x*** alpha characters/ ***y*** numeric characters

RSA EMC²

Token Policy *(cont'd)*

- Fixed Passcode
  - Option to use same settings as for PINs or define different settings
    - Lifetime, reuse, format
- Emergency Access Code Format
  - Options to include any or all of:
    - Numeric characters
    - Alpha characters
    - Special characters
  - Consider security policy and ease of use for Emergency Codes – e.g. if provided over phone, are randomly generated special characters easy to communicate

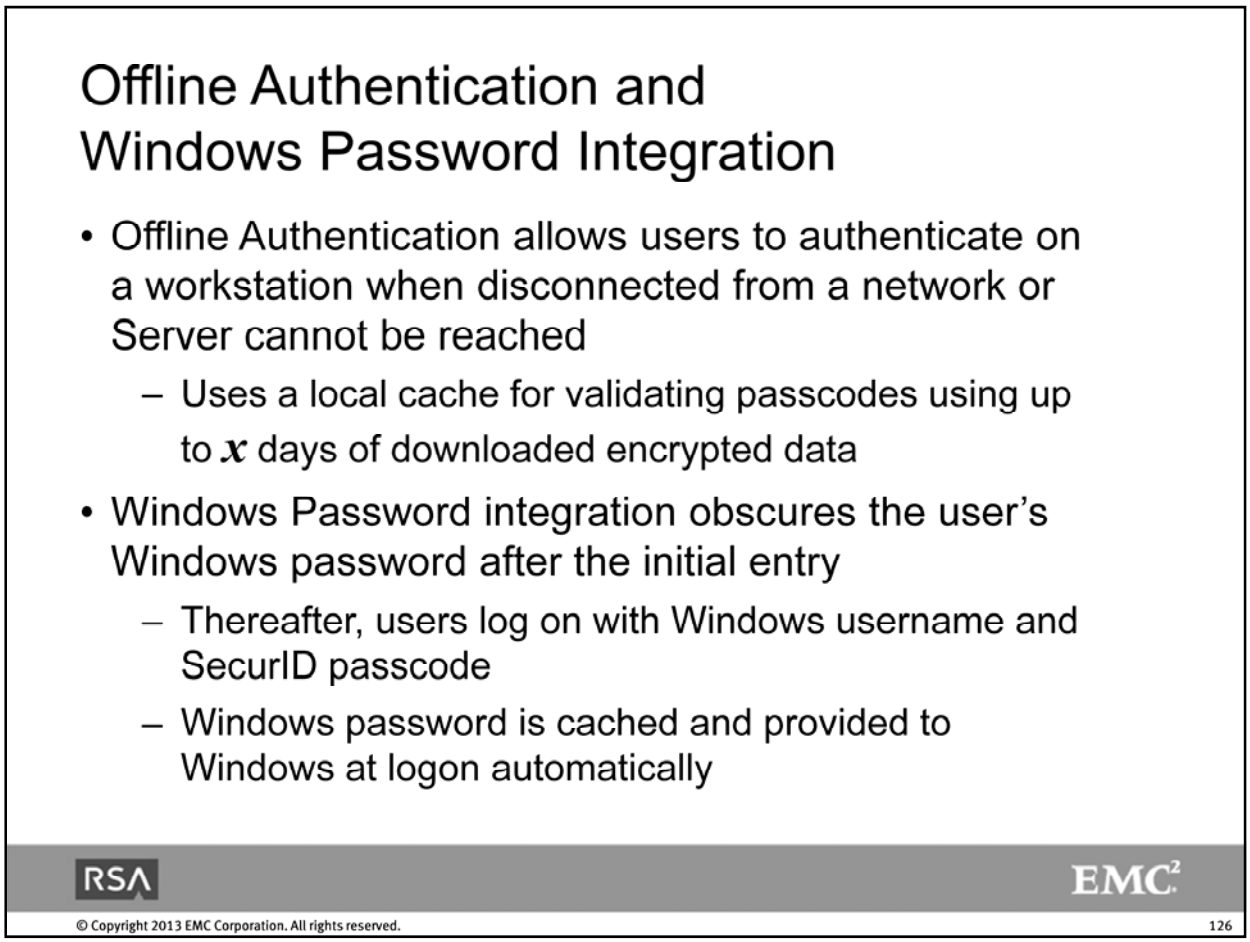
RSA EMC²

Exercise: Token Policy

- Establish policies for Token parameters

RSA EMC²

125



Offline Authentication

Offline authentication extends RSA SecurID authentication to users when the connection to RSA Authentication Manager is not available, for example, when users work away from the office, or when network conditions make the connection temporarily unavailable.

When offline authentication is enabled, Authentication Manager downloads a configurable number of "offline days" of tokencode data to users' machines. This data is used when users attempt to authenticate offline.

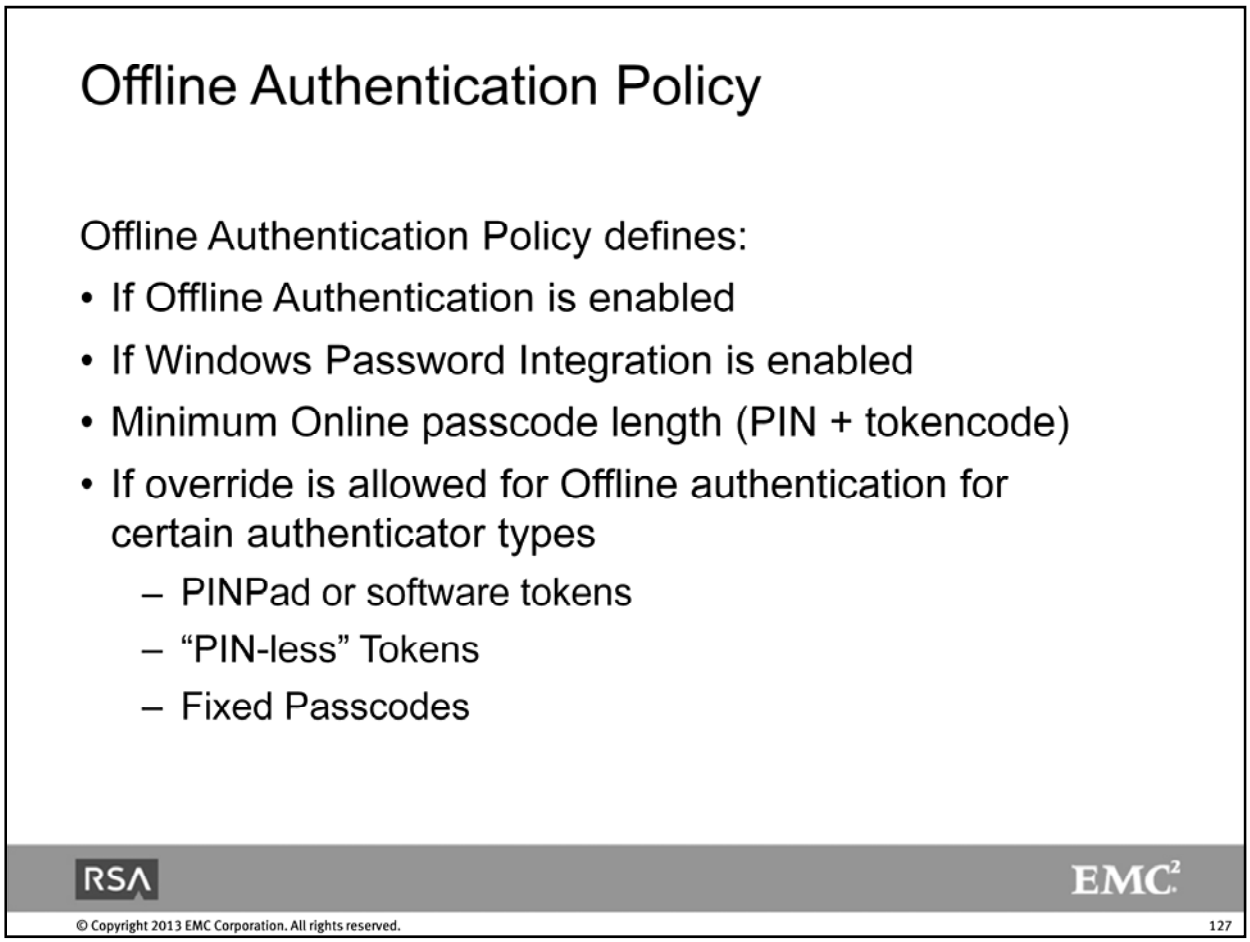
The number of offline data days is configured by the Offline Authentication Policy.

Windows Password Integration

Windows password integration integrates RSA SecurID into the Windows password logon process. Users provide their Windows logon passwords only during their initial online authentication. Passwords are then stored with the users' authentication data in the internal database and, for offline authentication, in the offline data.

During subsequent authentications, users enter only their user names and SecurID passcodes. The Authentication Agent gets the Windows password from the Authentication Manager or offline cache and passes it to the Windows logon system.

You enable local authentication and Windows password integration through the RSA Security Console, as part of an offline authentication policy. Only users assigned to security domains with an offline authentication policy that allows offline authentication and Windows password integration can use these features.



When Authentication Manager is installed, a default Offline authentication policy is automatically created. You can edit this policy, or create a custom policy and designate it as the new default policy. One offline authentication policy is always designated as the default policy.

When new security domains are created, Authentication Manager automatically assigns the default offline authentication policy to the new security domain. You can edit the security domain later to assign a new policy to replace the default policy.

Offline authentication policies assigned to upper-level security domains are not inherited by lower-level security domains. For example, if a custom policy is assigned to the top-level security domain, all new security domains created below it are still assigned the default offline authentication policy until changed to another policy.

Offline Authentication Policy *(cont'd)*

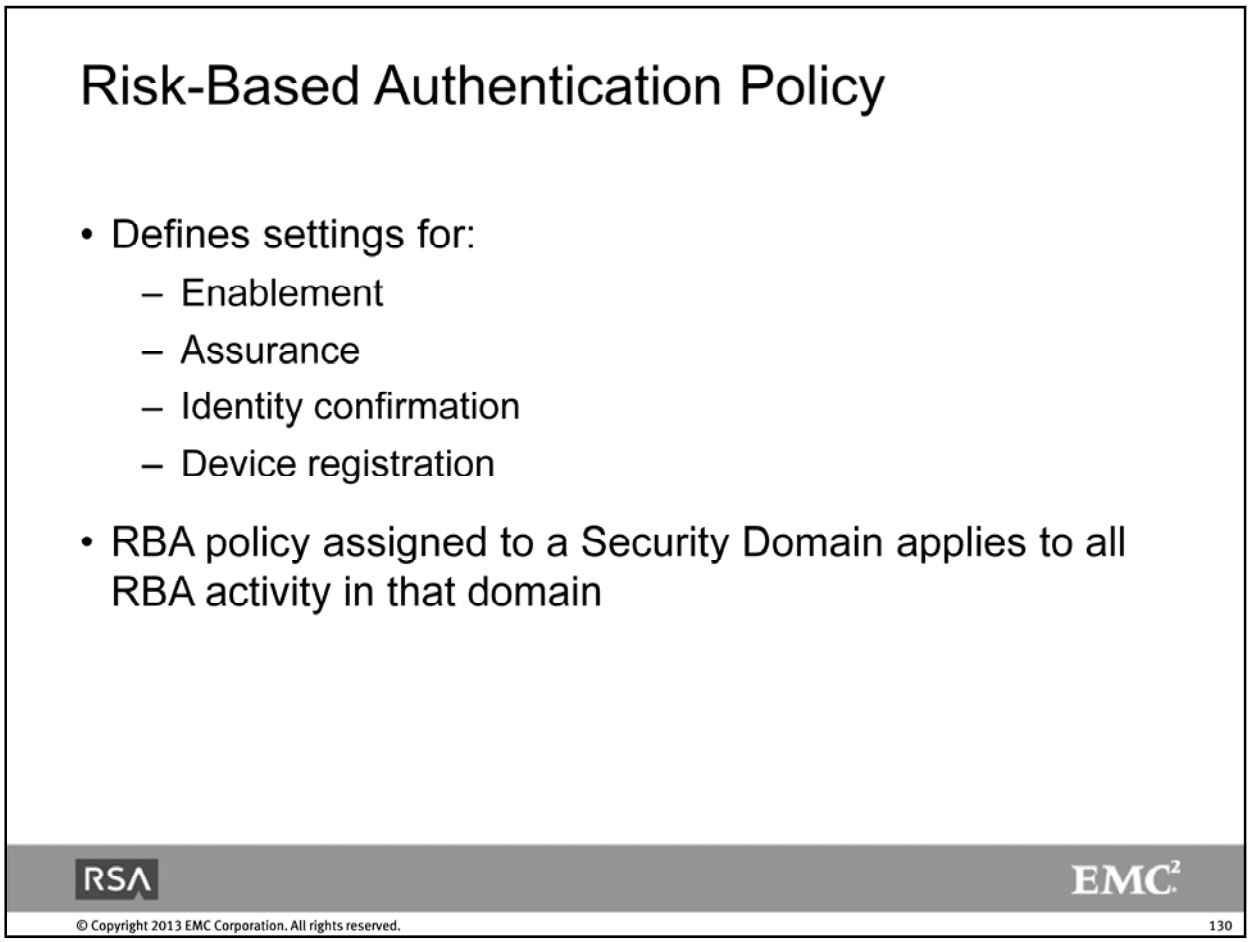
- If Offline Emergency Codes are allowed
  - Offline Emergency Tokencodes – used with PIN
  - Offline Emergency Passcodes – Replaces passcode
- Emergency Code lifetime
- Maximum # of days of offline data
- # of days for offline data warning
- Maximum offline failures (before user requires use of an emergency code)
- Enable offline logging – uploaded to Auth. Mgr when user reconnects to network

RSA EMC²

Exercise: Offline Authentication Policy

- Establish a policy for Offline Authentication

RSA EMC²



Risk-Based Authentication policies define how users are enabled for Risk-Based Authentication, assurance level to be used within the Security Domain to which the policy is assigned, how identity will be confirmed if additional authentication is required, and device registration.

Risk-Based Authentication Policy *(cont'd)*

- Enablement:
  - User must be enabled for RBA for the system to collect device information
  - Option to allow system to automatically enable users for RBA during authentication
- Minimum Assurance level:
  - High | Medium-High | Medium | Low
  - Default = 'Medium'
  - Probably requires security policy discussion, objectives for identity assurance, and experience to set the "correct" level

RSA EMC²

131

Risk-Based Authentication Policy *(cont'd)*

- Device Registration:
  - Establish Silent Collection period (default is 14 days after first successful authentication)
  - Option to not allow Silent Collection (users always need to authenticate on an unregistered device)
- Identity Confirmation:
  - Sets confirmation method as Security Questions **and/or** On-Demand Authentication
    - User must have email/mobile phone in profile to receive ODA
- New Device Registration
  - Can register user device automatically after authentication
  - Or--- Prompt the user to allow device registration

RSA EMC²

Risk-Based Authentication Policy *(cont'd)*

- Device Administration:
  - Sets maximum number of devices (per user)
    - Default = 20
- Unregister a device:
  - Removes device registration if not used in ***x*** days
    - Default = 60

RSA EMC²

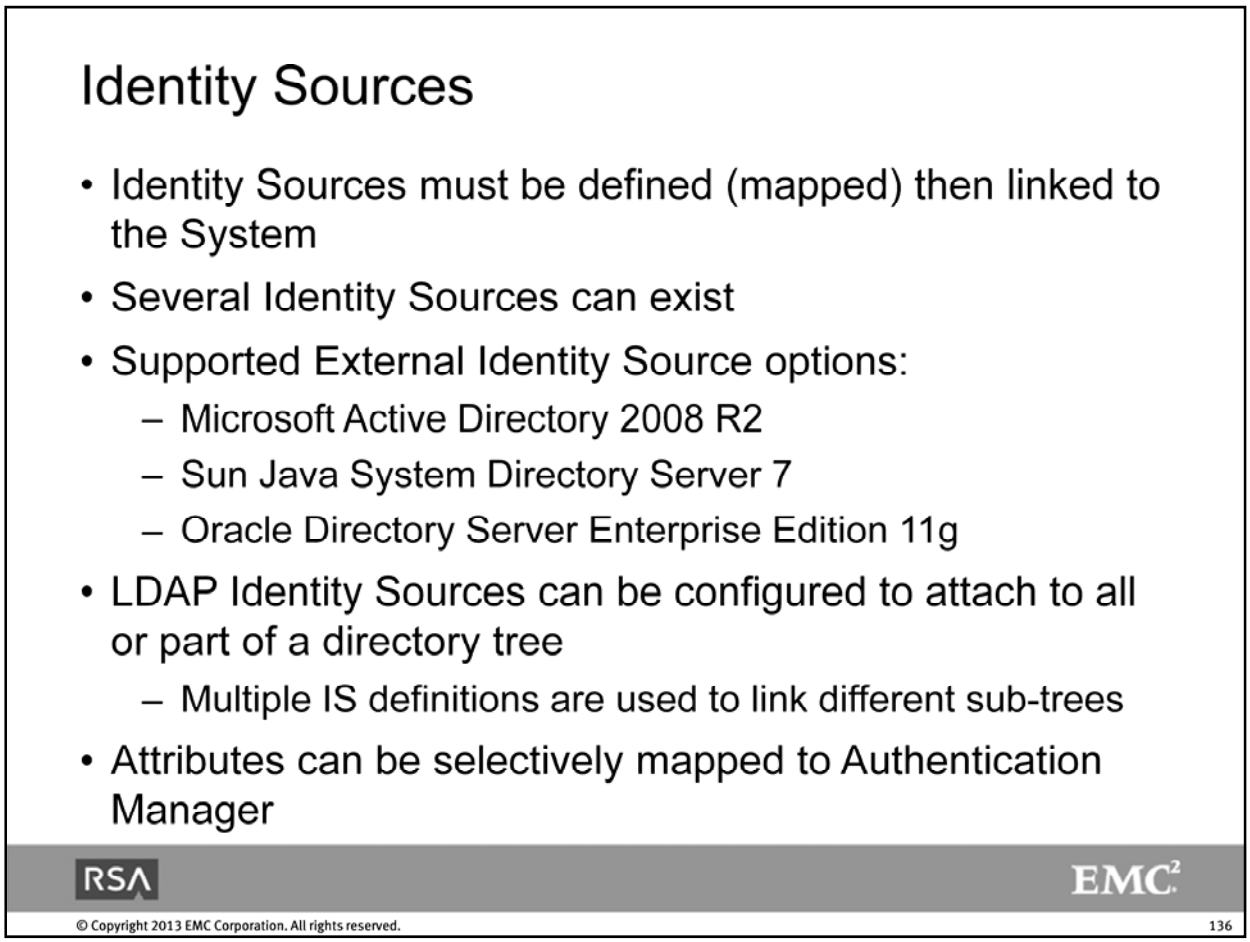
Exercise: Risk-Based Authentication and RBA Message Policy

- Establish policies for Risk-Based Authentication

RSA EMC²

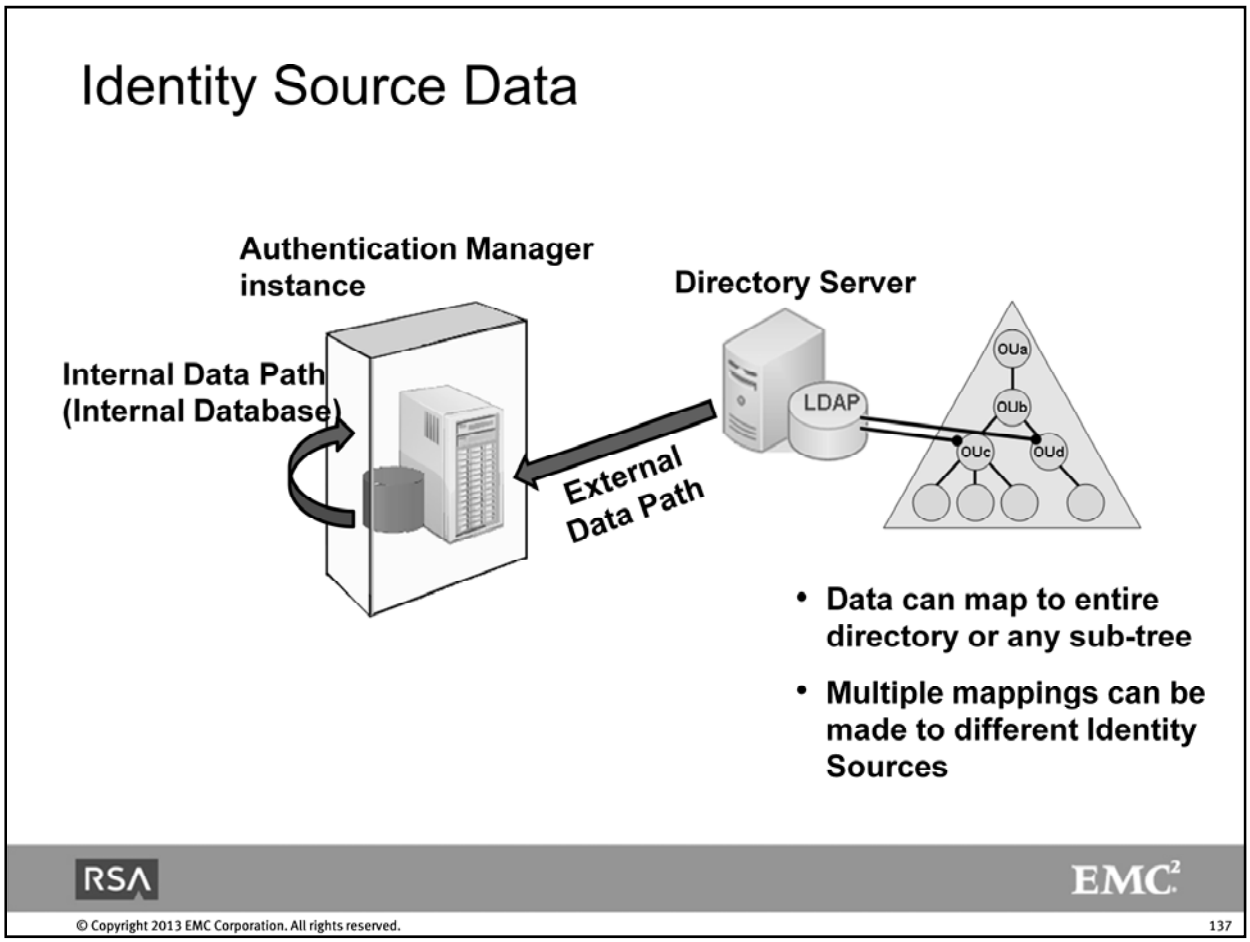
Identity Sources

Unit 6



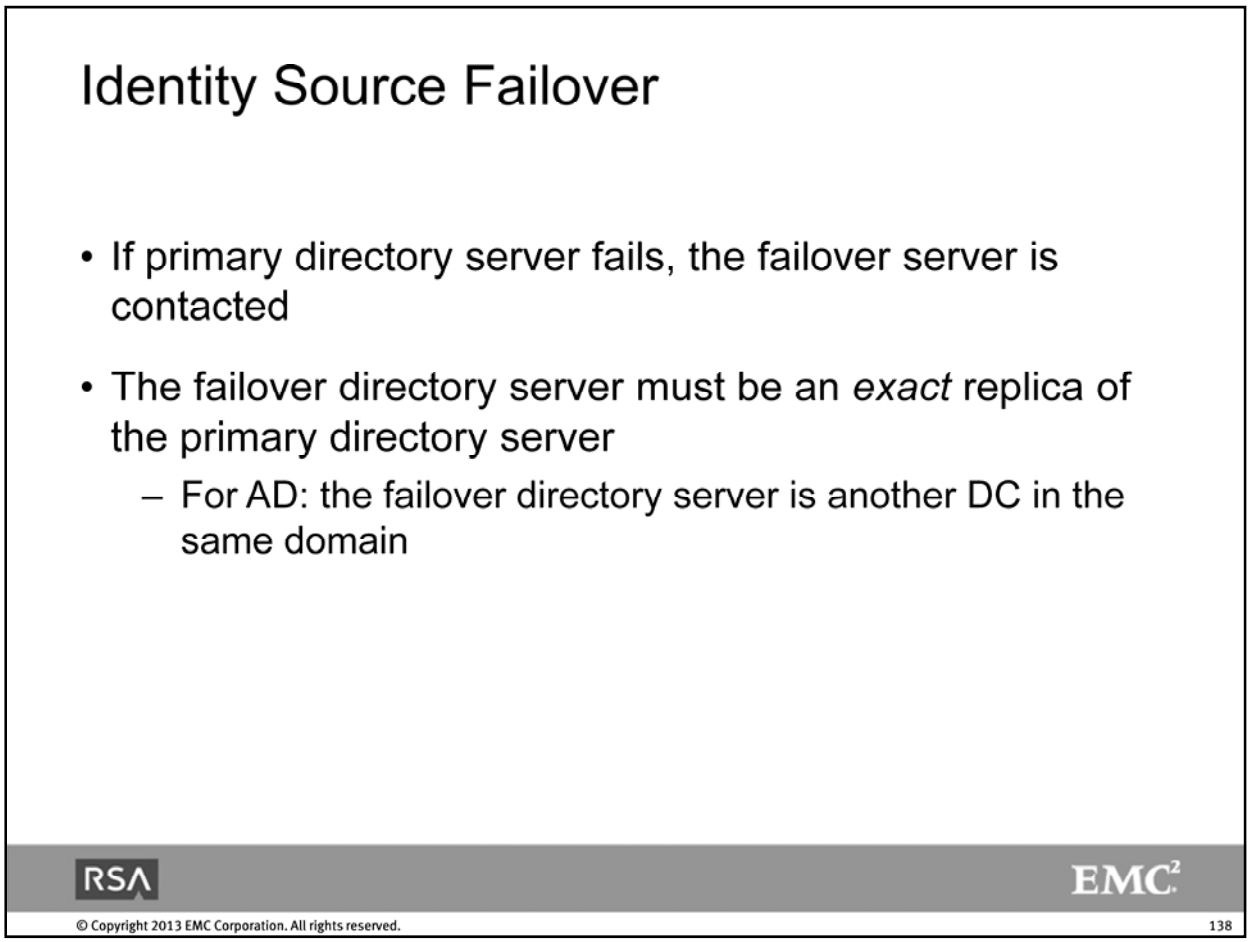
Authentication Manager enables the integration of LDAP data sources without modifying the LDAP schema. Administrators use the RSA management consoles to create an identity source, link it to Authentication Manager, and map Authentication Manager attributes to the LDAP attributes.

Multiple Identity Sources can be defined and linked to Authentication Manager.



When defining an Identity Source, you can specify all or part of a directory tree by indicating the Organizational Unit (OU) container.

Multiple Identity Source definitions - each using a different portion of the directory tree - can be used to attach to different OU segments without integrating the entire tree into Authentication Manager. If this approach is followed, however, the Identity Sources should not be configured so that portions of the tree overlap because it is possible to link duplicate objects.



When an Identity Source is defined, an additional failover directory server can also be defined. This failover server must be a replica of the primary directory server - using the same schema, the same general content and structure, and the same access credentials.

In the case of Active Directory, the failover directory server is another Domain Controller in the same domain.

Exercise: Defining an Identity Source

- Define Active Directory Identity Sources and link to the system

VARIABLES:
<AD_hostname>
<AD_Directory_passwd>

RSA EMC²

139

Identity Attribute

- An attribute associated with user records that is not provided out of the box
- Stored in either:
  - Identity Source
  - Internal Database (visible to all Identity Sources)
- Supported data types
  - String, Integer, Boolean, Date, Float
- Single and multi-values
  - With support for pre-defined value choices

RSA EMC²

Mapping the fields in an identity source to the fields in the Authentication Manager Security Console allows you to use the Security Console to view user and user group data stored in the identity source.

In addition to the default fields contained in a user record, you can define custom attributes, called identity attribute definitions, that contain information tailored to your organization.

For example, you might decide to define an attribute named “Department” in which you can enter a user’s department name, such as “HR” or “Finance”; or map a telephone number to help the Help Desk with security verification or other uses.

String, Integer, Float, Date, and Booleans data types are supported in Authentication Manager and additionally, you can pre-define value choices when creating an Identity Attribute.

When you define a custom attribute, the attribute definition is stored in the Authentication Manager internal database. By default, the attribute value is also stored in the internal database.

Identity Attributes can also be grouped by category and displayed or used as user search criteria.

Examples

- Employee ID (integer)
- Phone Number (string)
- Address (string)
- isSingleValued Flag (boolean)

Authentication Manager Attribute Usage

- Values are mapped for each Identity Source
  - No need to modify the schema
  - Each Identity Source can have its own mapping
- Can be restricted by role permissions
- May be displayed and/or used as criteria in user searches
- Can be grouped by Category
- Can be selectively hidden from the console

RSA EMC²

142

Exercise: Configuring Identity Attributes

- Configure Identity Attributes

RSA EMC²

143

Security Domains

Unit 7

Security Domains

- A security domain is a container that defines an area of administrative responsibility
- Security domains can be organized in a hierarchical tree
  - For example:

RSA EMC²

145

Security domains serve as containers that represent areas of administrative responsibility, typically business units, departments, partners, and so on. Security domains establish ownership and namespaces for objects within the system. All Authentication Manager objects are managed by a security domain.

Security domains allow you to:

- Organize and manage your users.
- Enforce system policies.
- Limit the scope of administrators' control by limiting the security domains to which they have access.

Security domains are organized in a hierarchy within a realm. You can create as many security domains as your organization requires. Security domains are often created to mirror the departmental structure or the geographic locations of an organization.

Security Domains *(cont'd)*

- All RSA managed objects are "stored" in security domains
- Security Domains are used in conjunction with roles to limit what is visible to an administrator and the operations they can perform on the visible objects
- Security domains "own" all RSA managed objects that they contain:

- In turn, all objects are owned by some Security Domain.

RSA EMC²

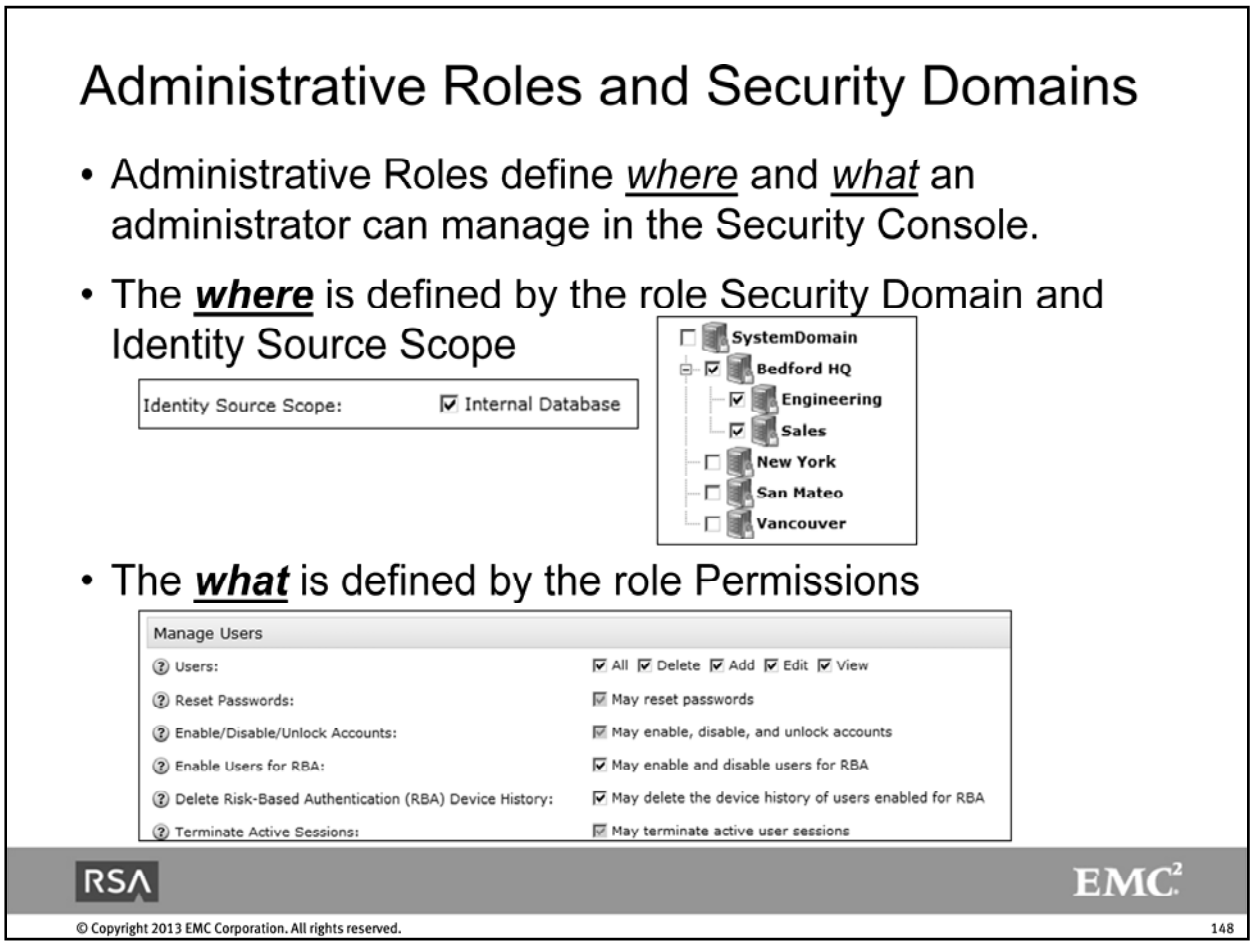
146

The combination of administrative roles, the security domain within which objects reside, and the scope allowed for an administrator can result in a complex logical arrangement so that a deployment properly divides and separates administrative responsibility while still allowing administrators to perform their required tasks.

Security Domains *(cont'd)*

- Security Domains contain nearly everything that can be managed through the Security Console
  - Tokens, principals, groups, agents, admin roles themselves
- They do NOT contain:
  - System-wide settings
    - Configuration data, identity sources, realm trusts, etc. *(Only accessible to super-admin)*

RSA EMC²



An administrator's role defines the objects and security domain that an administrator can manage in Authentication Manager. The security domain provides the boundaries of authority while the scope of authority is defined by the Identity Source.

When you assign permissions to a role, keep in mind that an administrator in that role might need to associate two objects in the deployment - such as Tokens to Users. The administrator must have the appropriate permissions and scope for both objects at both ends of the association. For example, to move users between security domains, an administrator must be able to view security domains and users.

The scope of an administrative role controls where an administrator may perform specified administrative tasks. A part of that scope consists of the portion of the security domain hierarchy that the administrative role can manage.

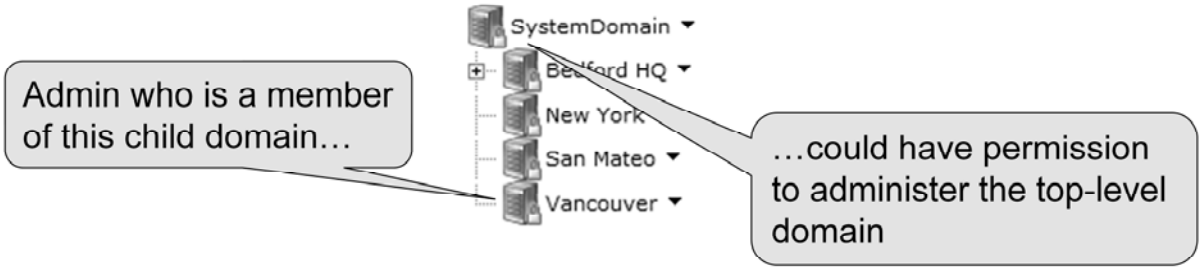
The strategy in creating security domains must take into account a balance of administrative scope - broad enough so that the administrator can access all the necessary security domains and identity sources but not so unnecessarily broad as to grant access to security domains where the administrator should have no responsibilities.

For example, a Help Desk Administrator can edit the user record of any other administrator within his scope. This means that a Help Desk Administrator can change the password of a higher-level administrator and gain the administrative privileges of the higher-level administrator. To avoid such situations, you can assign the higher-level administrator to a security domain that is not within the scope of the Help Desk Administrator.

An administrative role manages only within the security domain where the role definition was saved. This includes all of the lower-level security domains in the same security domain. An administrative role can only manage down the security domain hierarchy, not up.

Security Domain Scoping

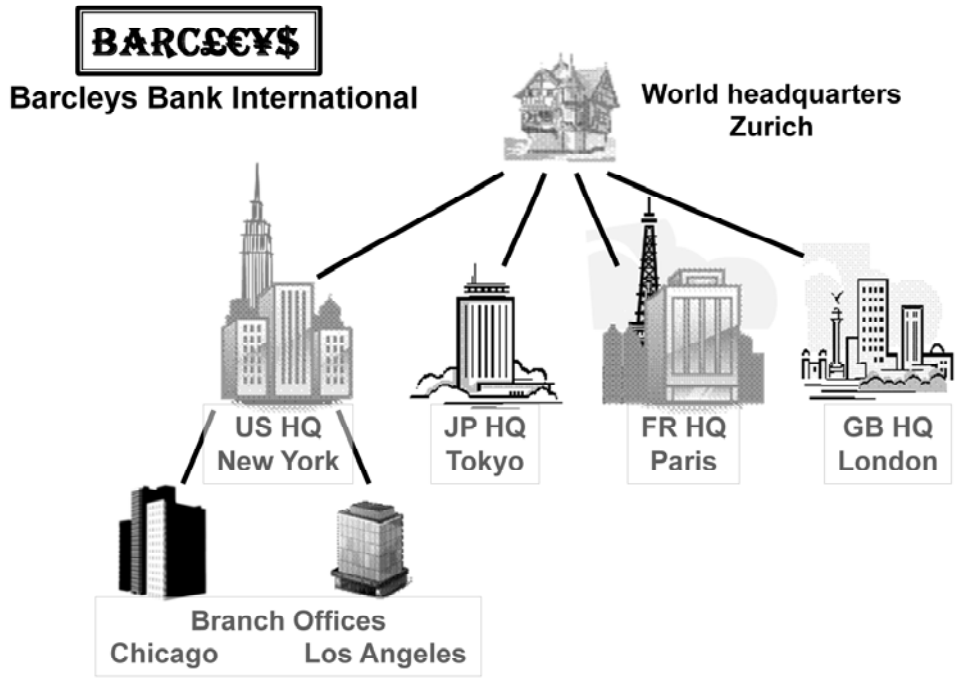
- Administrator's own security domain is unrelated to the role's scope
  - Admin account could be saved at the lowest level but have role granting scope for all levels
  - Can even have permissions to manage self, or own admin role



RSA EMC²

149

Scenario for Security Domain Exercise



RSA EMC²

150

Exercise: Structuring Security Domains

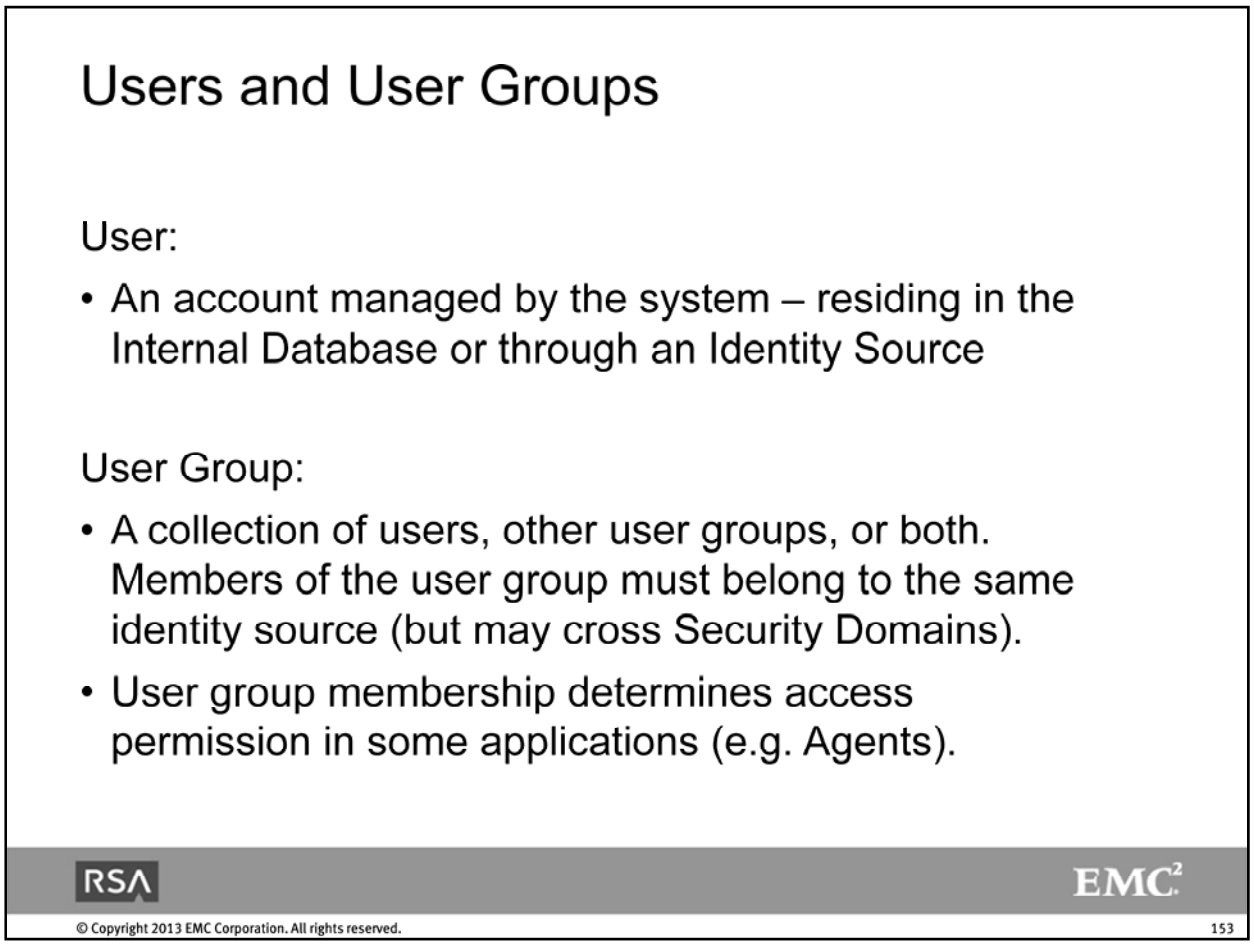
- Define a Security Domain hierarchy for your Authentication Manager deployment

RSA EMC²

Managing Users and User Groups

Unit 8

RSA EMC²



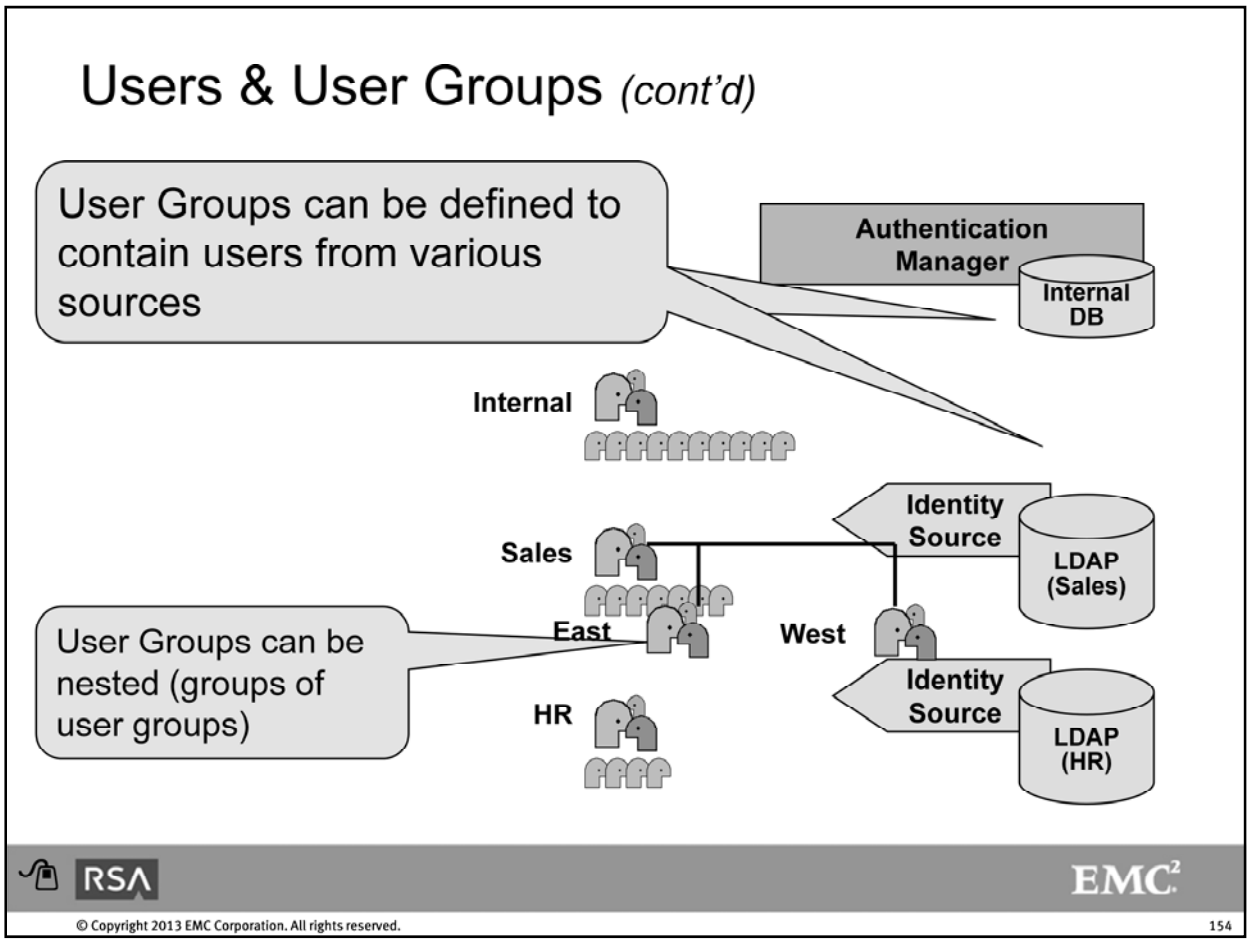
In RSA Authentication Manager terminology, a User is an account that is managed by Authentication Manager. Typically, a user account represents a person but it can also represent a computer entity, service -such as a web or system service, etc.

A user account is accessed through the RSA Security console and the degree of control and management depends on where the user account resides. If the account resided in the internal database, the administrator has full control over the user (add, delete, enable/disable, enforce password policy, etc.). If the user resides in an LDAP source and the identity source is established as read-only, the Authentication Manager administrator can only control the association of the user account with other entities (such as assignment to a token) but not fully control the account itself. In such a case, the Authentication Manager administrator will need to cooperate with the LDAP administrator to coordinate user account management when required.

A User Group is a collection of users, other user groups, or a combination of both types of objects.

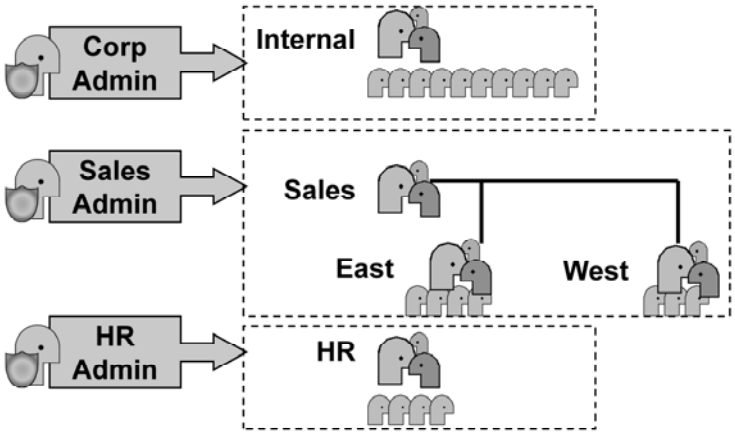
Members of a user group must belong to the same identity source but within one Identity Source, members can consist of a mix of users from one or more Security Domains.

Membership in a group determines access privileges in some applications (for example, activation on a restricted Agent) and can be used to apply group policies such as restricted access times.


Users & User Groups (cont'd)
User Groups can be defined to contain users from various sources
Authentication Manager
Internal DB
Internal
Identity Source
LDAP (Sales)
Sales
East
West
Identity Source
LDAP (HR)
HR
User Groups can be nested (groups of user groups)
RSA
EMC²
© Copyright 2013 EMC Corporation. All rights reserved.
154

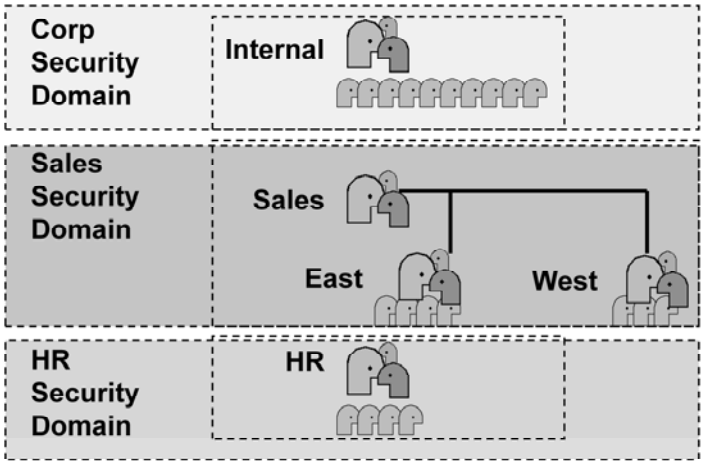
Users & User Groups *(cont'd)*

Administrators can be given group scope to manage specific sets of users



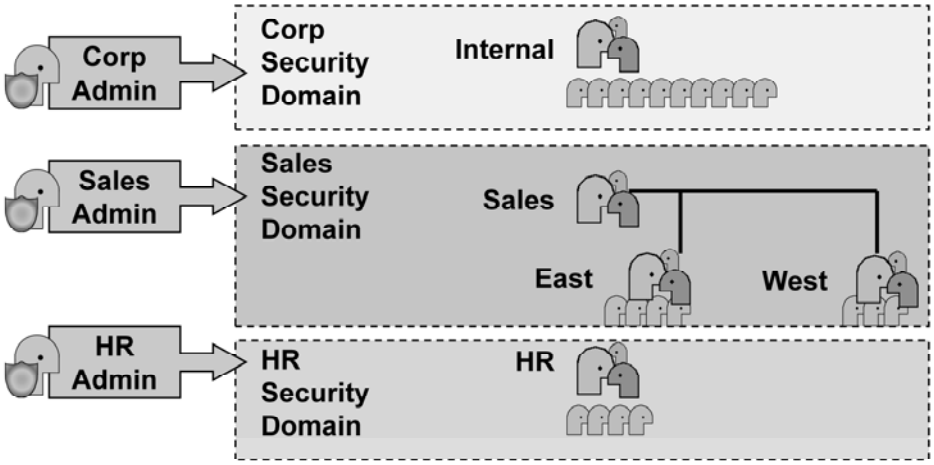
Users & User Groups *(cont'd)*

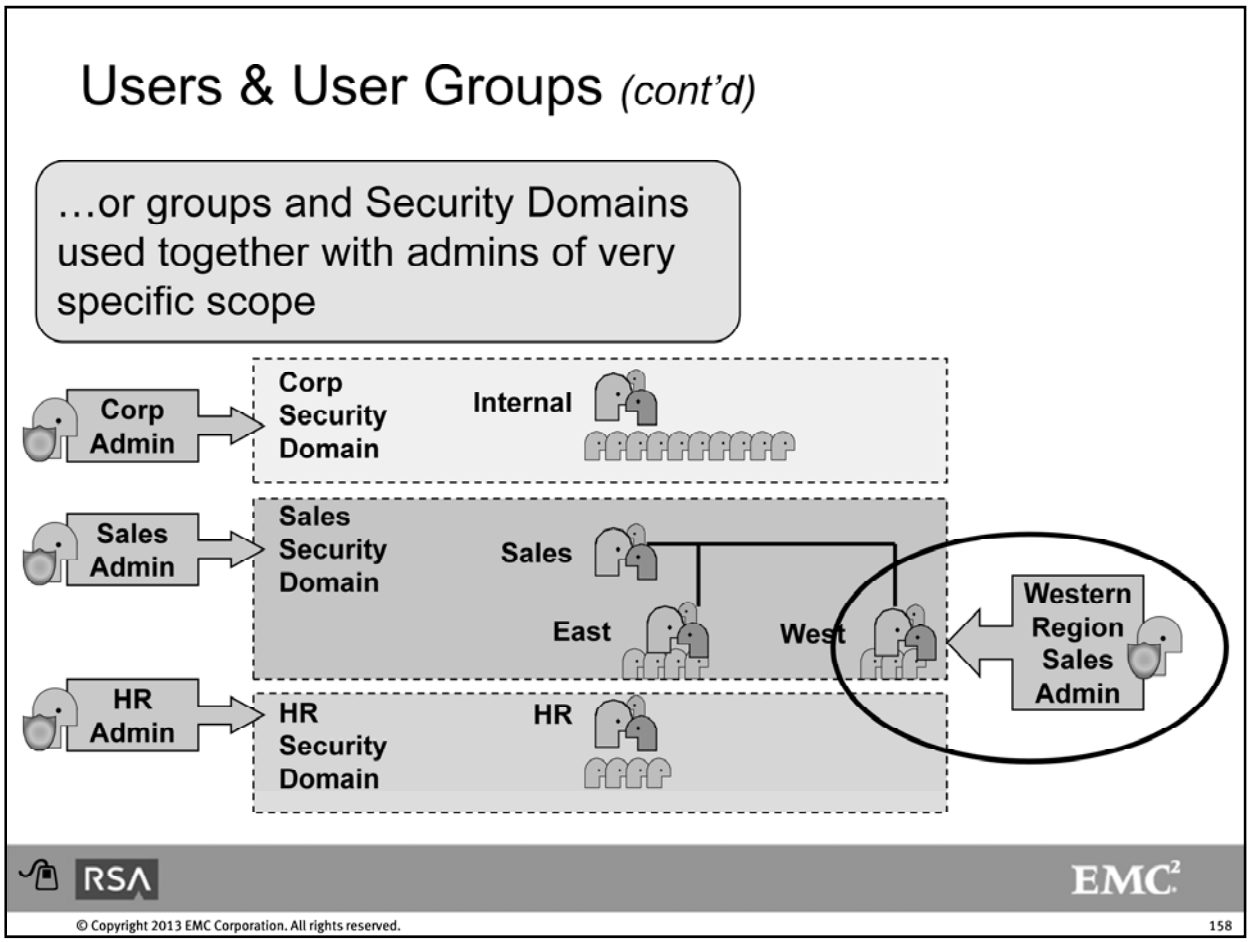
Security Domains could also be used to organize like groups of users…

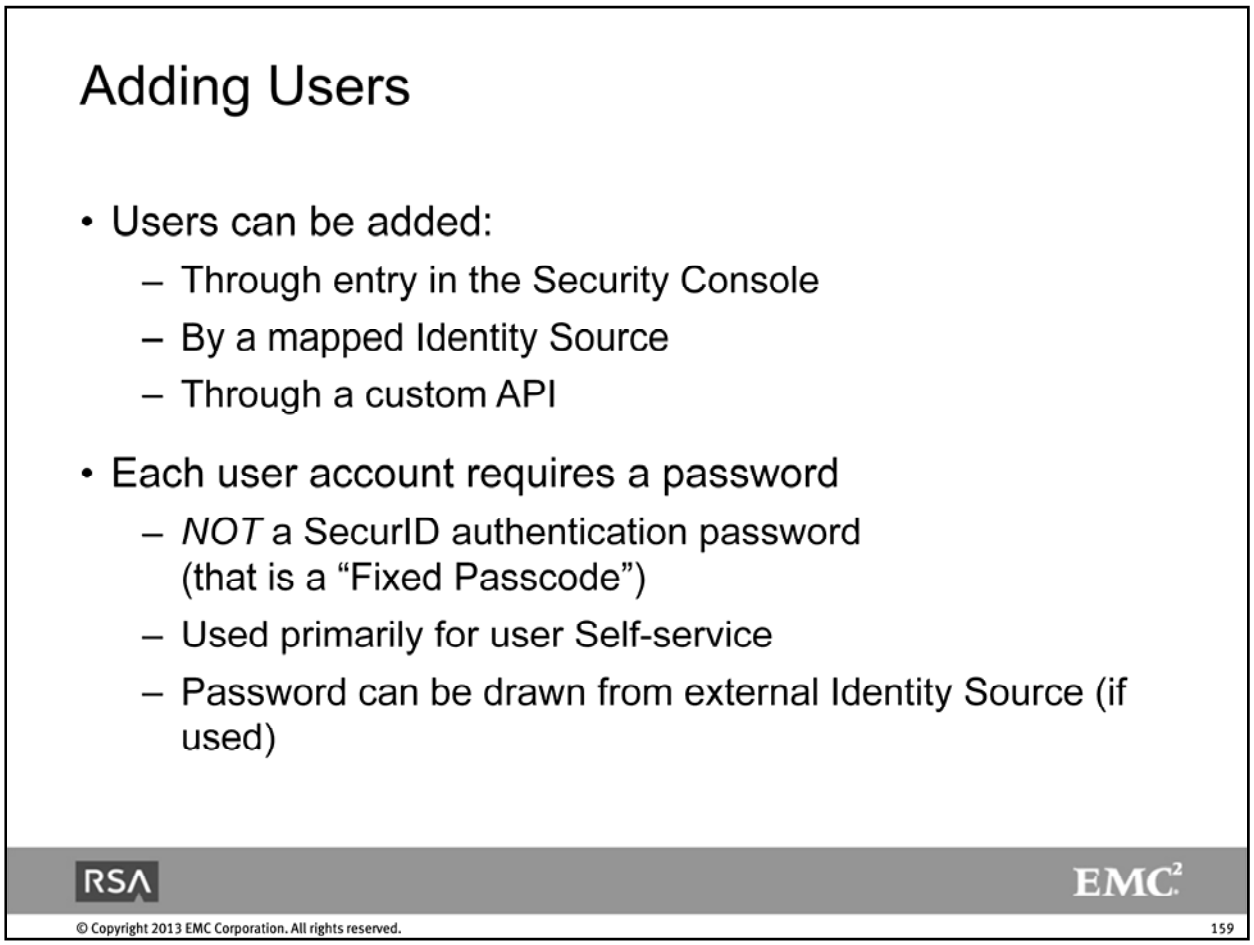


Users & User Groups *(cont'd)*

…and Administrators assigned with scope over the Security Domains…




Users & User Groups (cont'd)
…or groups and Security Domains used together with admins of very specific scope
Corp Admin
Corp Security Domain
Internal
Sales Admin
Sales Security Domain
Sales
East
West
Western Region Sales Admin
HR Admin
HR Security Domain
HR
RSA
EMC²
© Copyright 2013 EMC Corporation. All rights reserved.
158



User accounts can be added to Authentication Manager in three ways:

- Through direct entry through the Security Console
- By mapping through an Identity Source
- By using a custom API application to add users from some external datastore or file

Each user account requires a last name, userID, and password. The password of the user account is used for activity within Authentication Manager (access to the Self-service console; administrative access if the user is an administrative user and does not require other authentication methods to access the Security Console).

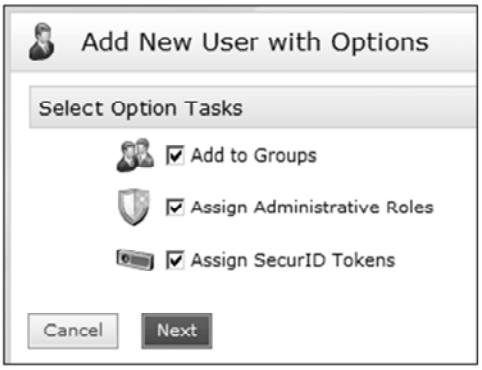
The password that is established in the user account is not a password that is used when the user is prompted for SecurID authentication - it is not a SecurID "Fixed Passcode".

User accounts can be established with Start and Expiration dates - for example, when creating an account for a temporary employee.

The Security console offers the capability to 'Add User with Options' to allow assigning group membership, administrative roles, or SecurID tokens in a chained series of operations.

Adding Users *(cont'd)*

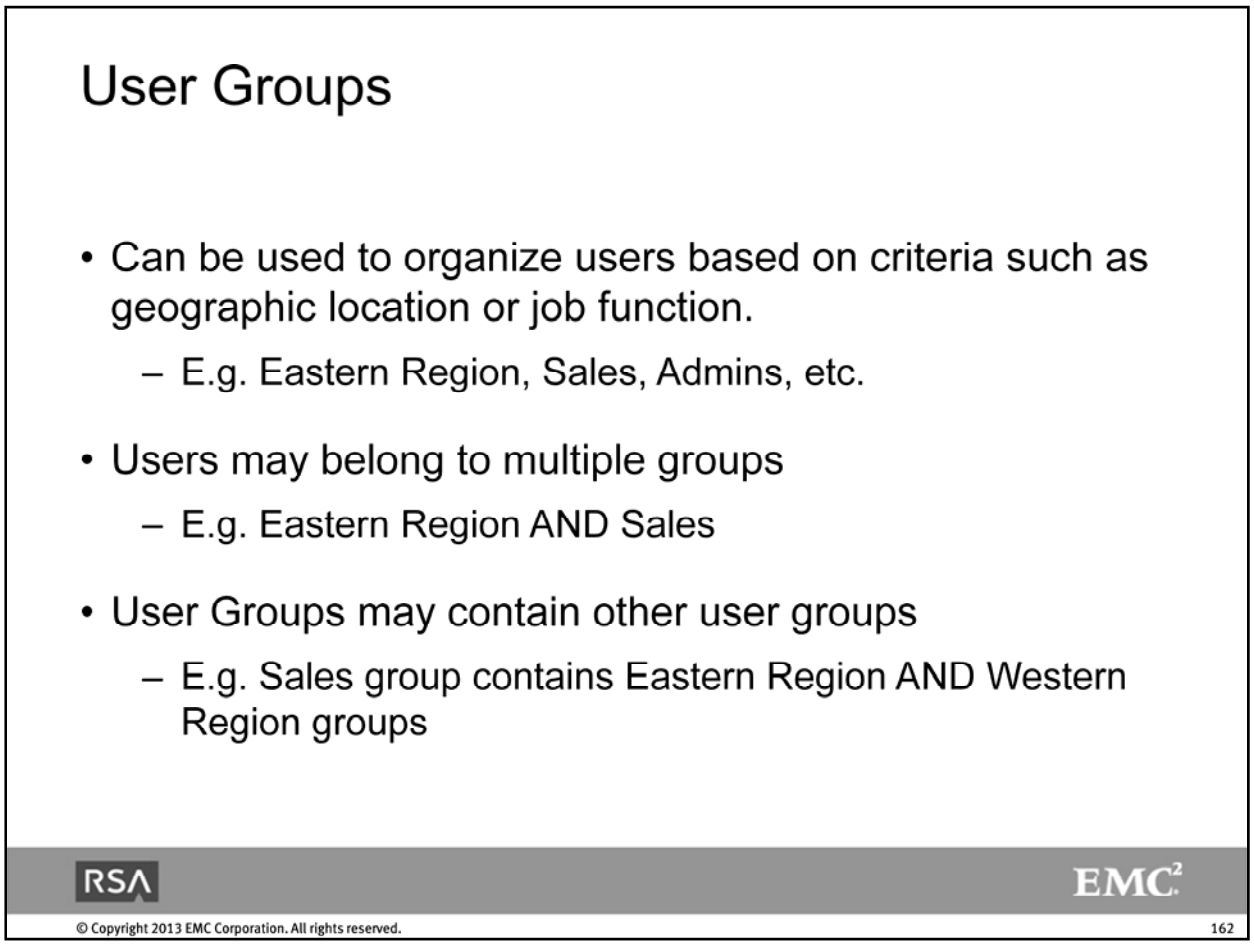
- Users can be established with an account start and expiration date (for example, a temporary employee)
- 'Add User with Options' allows you to perform multiple tasks during one Add operation (acts like a Wizard)



Adding Users *(cont'd)*

- If Identity Attributes exist, data can be supplied at the time of user creation (*must* be - if the attribute is 'required')
- Username must be unique within the system
- Users must reside in some Identity Source
- Users exist in only one Security Domain but can be moved between Security Domains (through Edit User function)
- User record can be duplicated to retain group memberships, attributes, etc. for a new user (only name and password needs to be added)

RSA EMC²



User Groups are used to group and organize users based on similar criteria. Typically, criteria such as geographic location or job function are used. (For example, a facility or regional office, a department such as Sales, an administrative group, etc.).

Users may belong to multiple groups (for example, the Sales group and the Marketing group).

A User Group can encompass other groups (for example, the Sales group can contain the Inside Sales group and the Field Sales group) as well as a combination of groups and individual user accounts.

Like user accounts, User Groups are stored within an Identity Source and are 'owned' by some Security Domain. The User Group itself as well as all group members must belong to the same Identity Source but can be part of multiple Security Domains.

User group names must be unique within an Identity Source but one single Realm may have groups with the same name as long as each group is part of different Identity Sources.

User Groups can be moved between Security Domains.

User Groups are used to allow access to applications such as restricted Agents. Group activation is the only method whereby users are granted access through an Agent (individual activation is not allowed). Access times applied to a group define a restriction of when users are allowed to authenticate to a given Agent (only during business hours, prevent night and weekend access, etc.)

Like user accounts, a User Group can be duplicated to retain its properties (user members, access times, Agent activations, etc.) and re-named to create a new group.

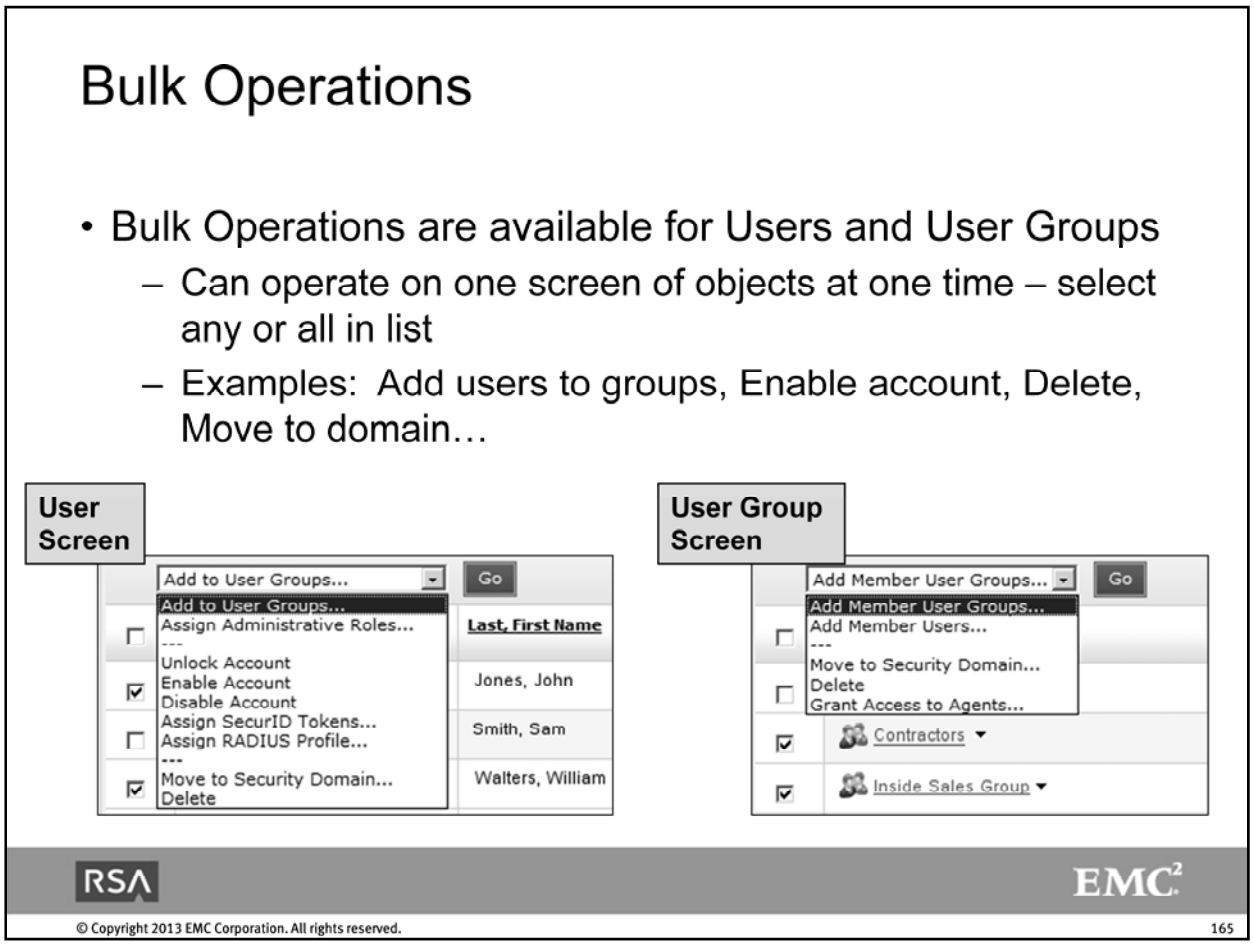
User Groups *(cont'd)*

- User groups are stored in an Identity Source and are owned by a Security Domain
- The user group and all user group members must belong to the same Identity Source but can cross Security Domains
- User group names must be unique within an Identity Source
  - It is possible to have multiple user groups with the same name if they are stored in different Identity Sources (but not a very good practice if it can be avoided)

RSA EMC²

User Groups *(cont'd)*

- User groups can be used to allow access to restricted Agents
  - *Only* group activation on Agent is allowed; *NOT* by individual user
  - Can specify access times that a group is allowed to access the Agent



Bulk operations - operating on several objects at a time - are allowed for User and User Group functions.

When accessing objects in the Security Console (**Users > Manage Existing**, for example) listing rows contain selection checkboxes. Selecting one or more rows allows you to operate on all selected objects (for example, select three users to be added to the Sales group).

Operations include adding individual users to a group, enabling/disabling account, deletion, moving to another domain, etc.

The only restriction to be aware of is that selections operate only on one screen of objects at a time. For example, if you want to add all users with last name beginning with 'S' to the S_Group, your screen lists 25 rows at a time and there are 40 users named S..., you will need to perform two bulk operations - once on each of two screens. If you need to perform bulk operations with a large number of users, you can increase the displayed row value or operate on certain sets of users by adjusting the search criteria.

Exercise: Managing Users and User Groups

- Add users to your deployment and perform user operations
- Add User Groups to your deployment and perform group operations

RSA EMC²

166

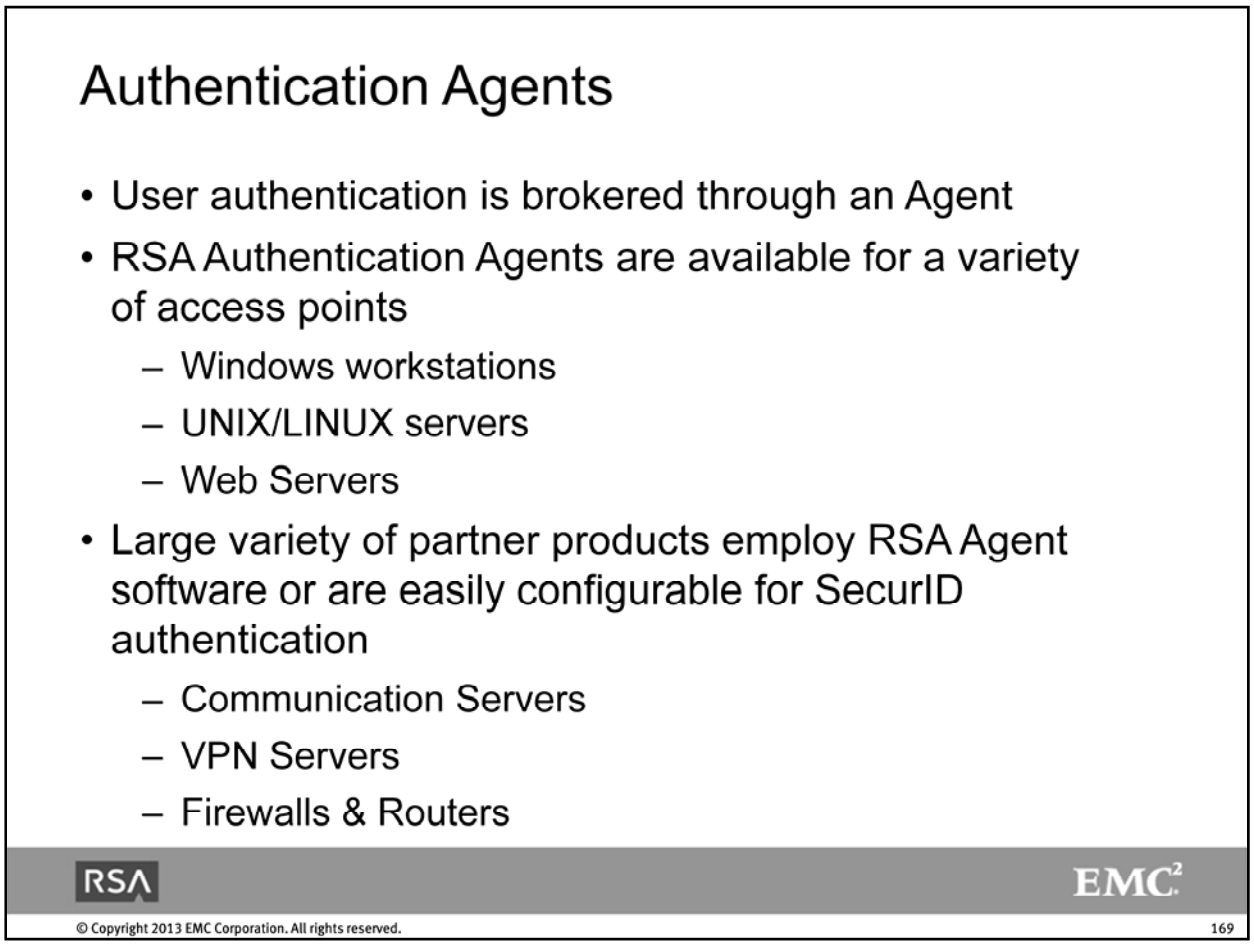
Exercise: I'm tired of Logging in to the Security Console

- Extend Security Console lifetime

RSA EMC²

Agent Operations

Unit 9



The RSA Authentication Manager authenticates a user's identity based upon the user's supplied Passcode. It is the job of the RSA Authentication Agent to obtain the user's Passcode and communicate with the server to verify that Passcode.

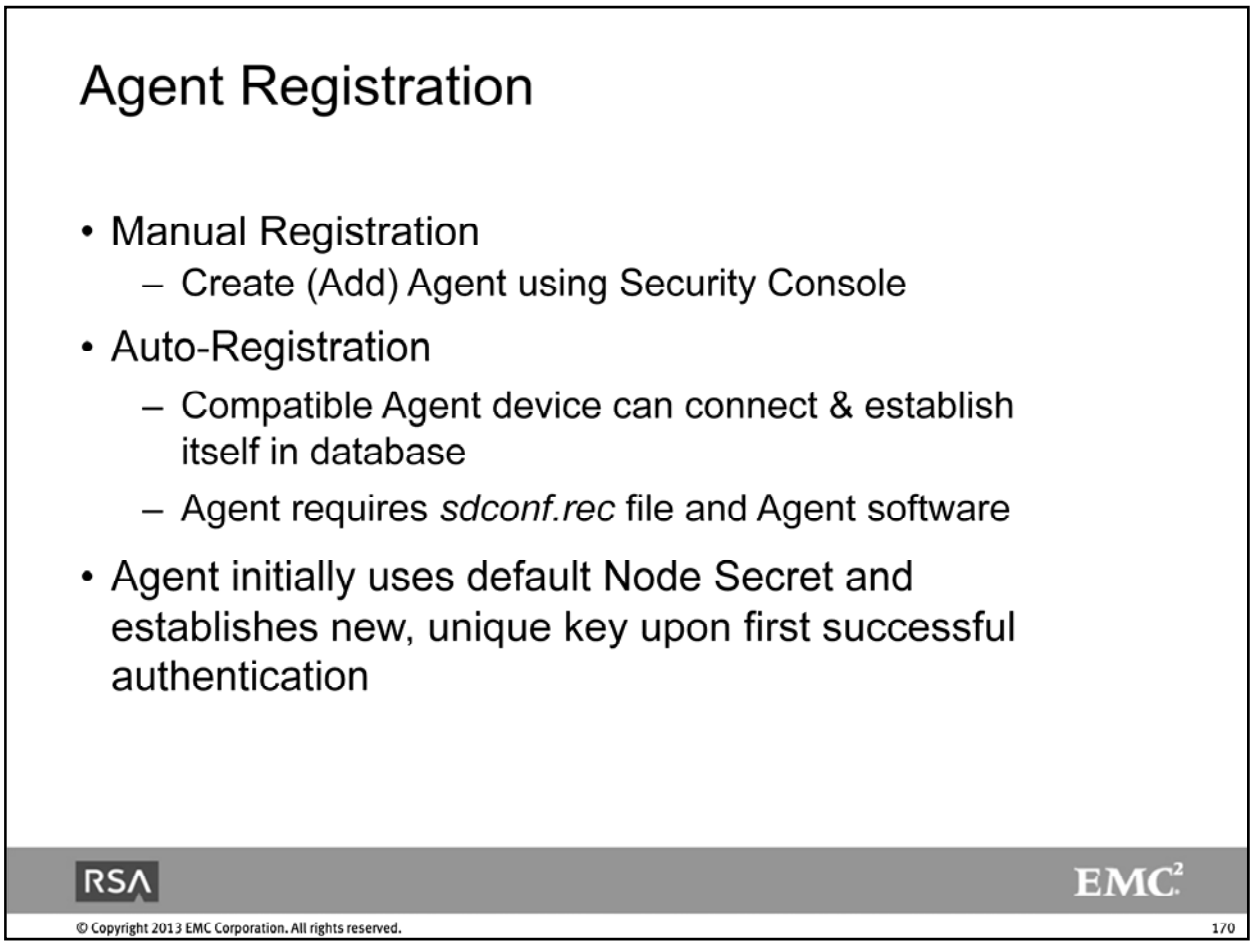
All user authentication is brokered through an Authentication Agent.

Protecting various system entry points or specific resources requires Agent software to be installed and/or activated. System or security administrators determine the network resources and entry points to be protected – often based upon an organization's Security Policy.

RSA Authentication Agents are available for a variety platforms.

In addition, a large variety of partner products employ RSA Agent software or are easily configurable for SecurID authentication. Such third-party partner products carry an identification "Secured by RSA".

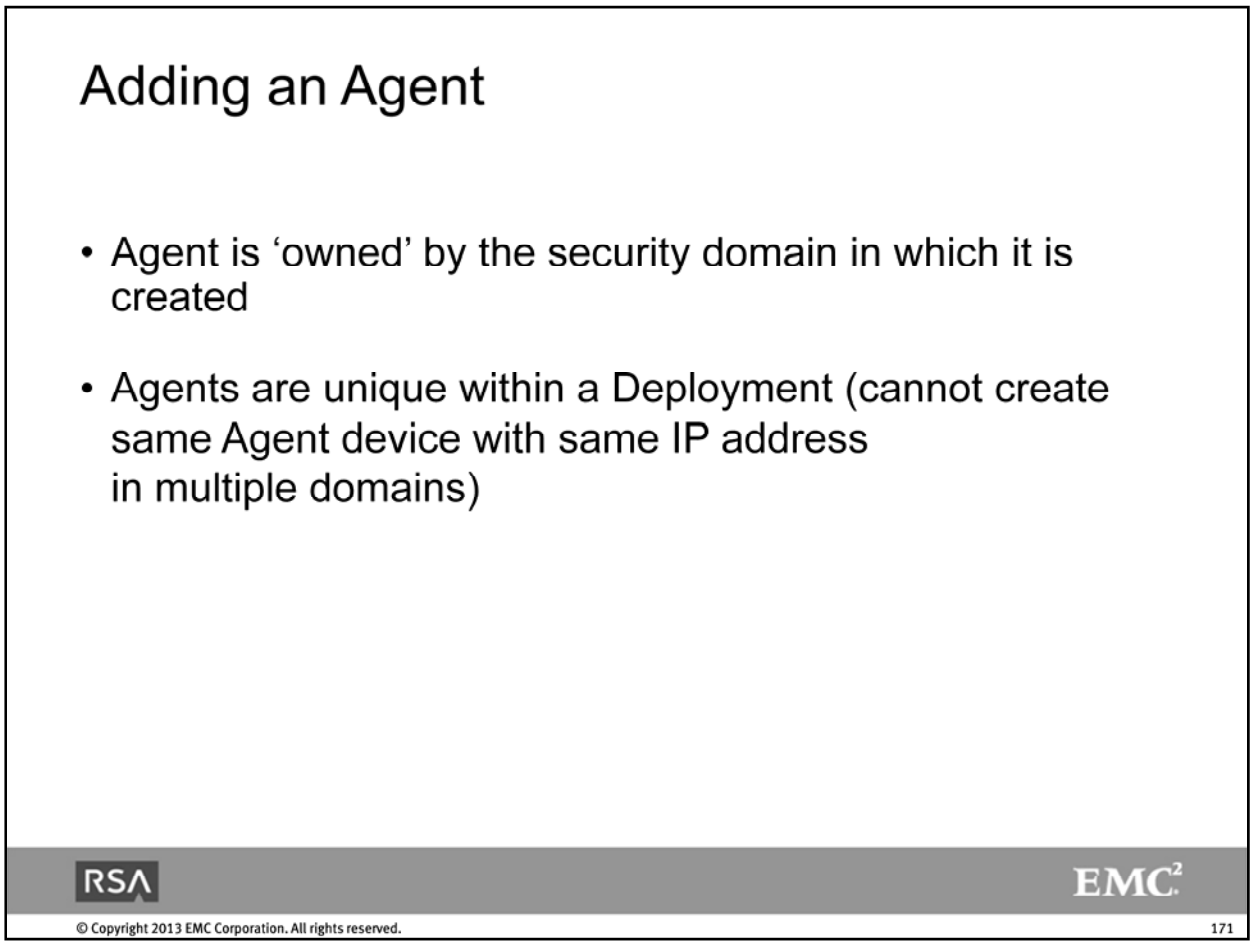
The RSA Web site and RSA support site (RSA SecurCare Online) always has a complete, up-to-date list of Agents, Secured by RSA partner devices as well as "Implementation Guides" on how to configure specific partner products.



Agents must be registered in the Authentication Manager database to be considered valid.

An Agent can also 'auto-register' itself in the database if the Agent software has auto registration capability, is supplied with an 'sdconf.rec' configuration file, and if Authentication Manager is configured to allow auto Agent registration.

The Node Secret key must match between Agent and Server to allow intelligible encryption/decryption of messages. Administrators can re-set the Node Secret if it becomes corrupt or the file containing the Node secret value is inadvertently deleted



Authentication Agents are added (registered) through the Security Console. Initially, the Authentication Agent is 'owned' by the security domain in which it is created - the security domain can be changed if it is advantageous to do so at a later time,

Because Agents must be unique within a Deployment (two Agents cannot have the same IP address) only a single instance is allowed across security domains .

Exercise: Managing Agents

- Add an Agent to your deployment
- Assign access to a User Group

RSA EMC²

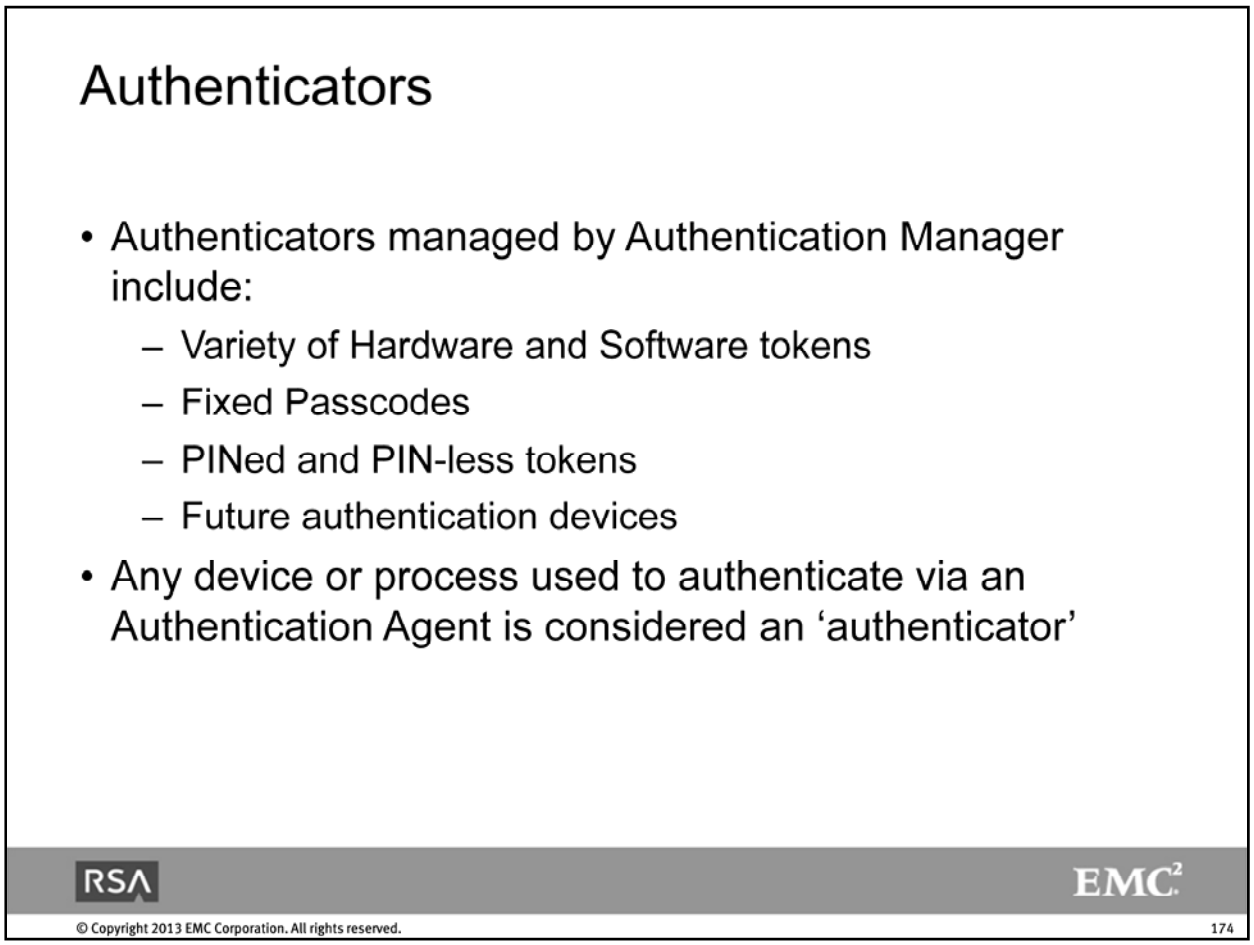
172

Authenticator Operations

Unit 10

RSA EMC²

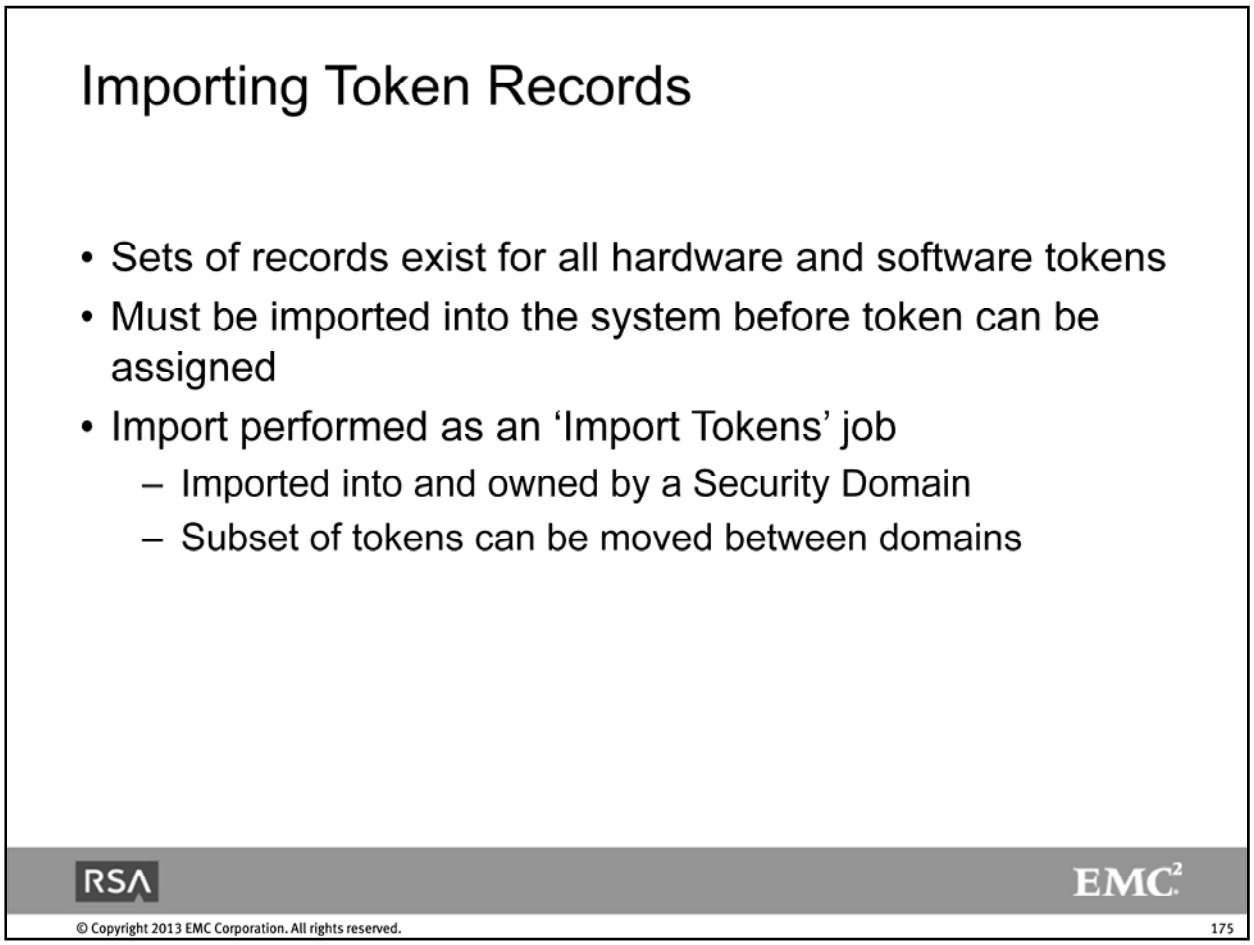
173



The range of authenticators that you can manage in RSA Authentication Manager include:

- Hardware and software tokens - including time-synchronous, event-synchronous tokens
- Fixed Passcodes
- Tokens used with and without PINs

In addition, RSA Authentication Manager has the capability to manage On-Demand tokencodes used for authentication and Emergency Access codes for both on-line and off-line access if a user cannot use their RSA SecurID token.



Each RSA SecurID Hardware or Software Token has a matching Token Record that must be imported into Authentication Manager. The record contains data such as serial number, algorithm and seed information.

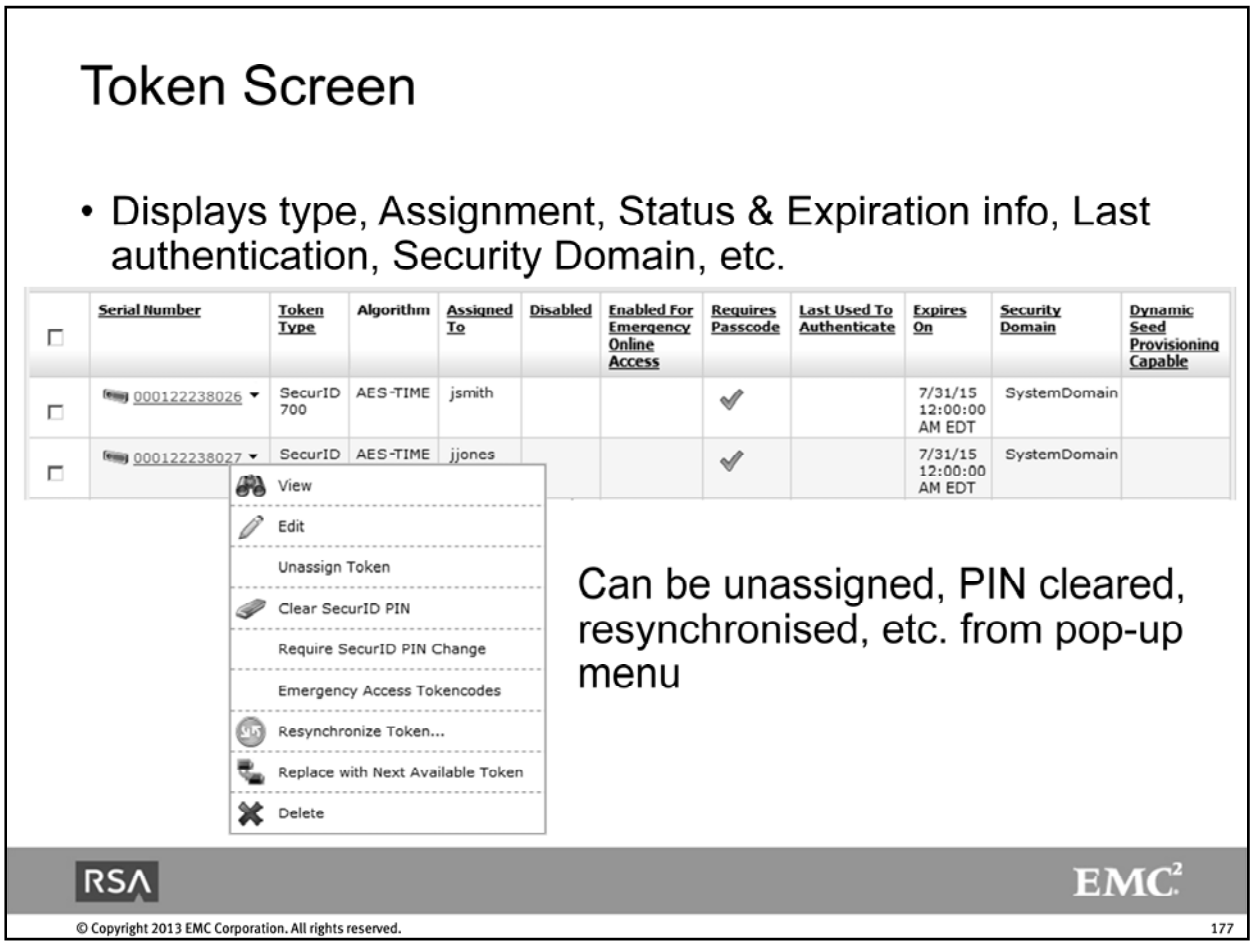
Importing a token record is performed through an 'Import Tokens' job through the Security Console.

Exporting Token Records

- Token records can be exported to another Authentication Manager deployment (or used to move Users & Tokens between Identity Sources)

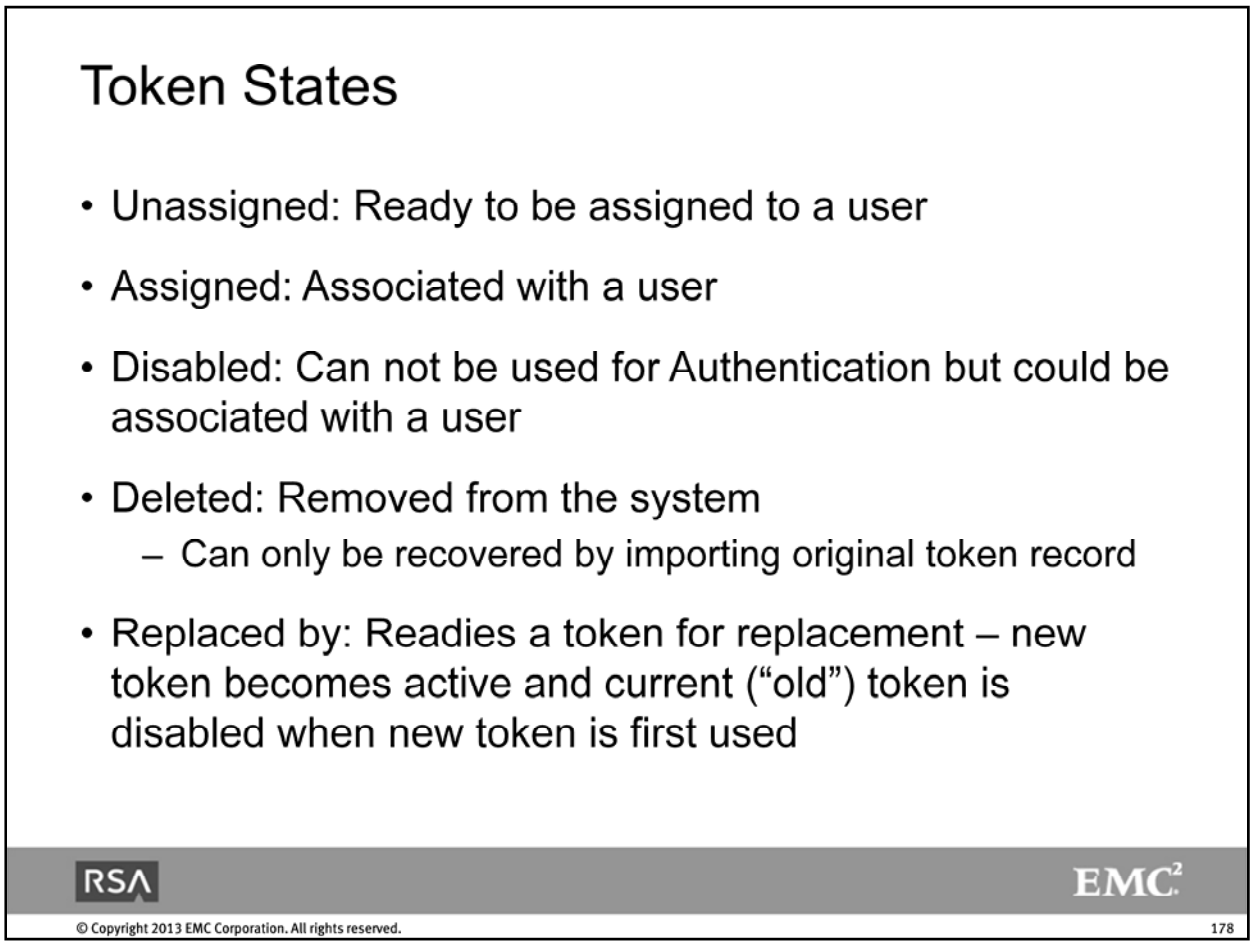
1. Obtain an encryption key from the target system
2. Choose 'Tokens Only' or 'Users with Tokens'
   - Can be isolated by Group membership
3. Choose Security Domain and status of source records
   - Status is 'All', "Assigned', or 'Unassigned'

RSA EMC²



The Token screen in the RSA Security Console displays basic information about the tokens in your system. The screen indicates:

- The Token Type
- Algorithm used by the token (AES or the older SID algorithm)
- If authentication requires a passcode (use with PIN) or not (PIN-less)
- The active state (enabled/disabled)
- Replacement information
- Expiration information
- Security Domain ownership
- Any special notes added by an administrator for a particular token



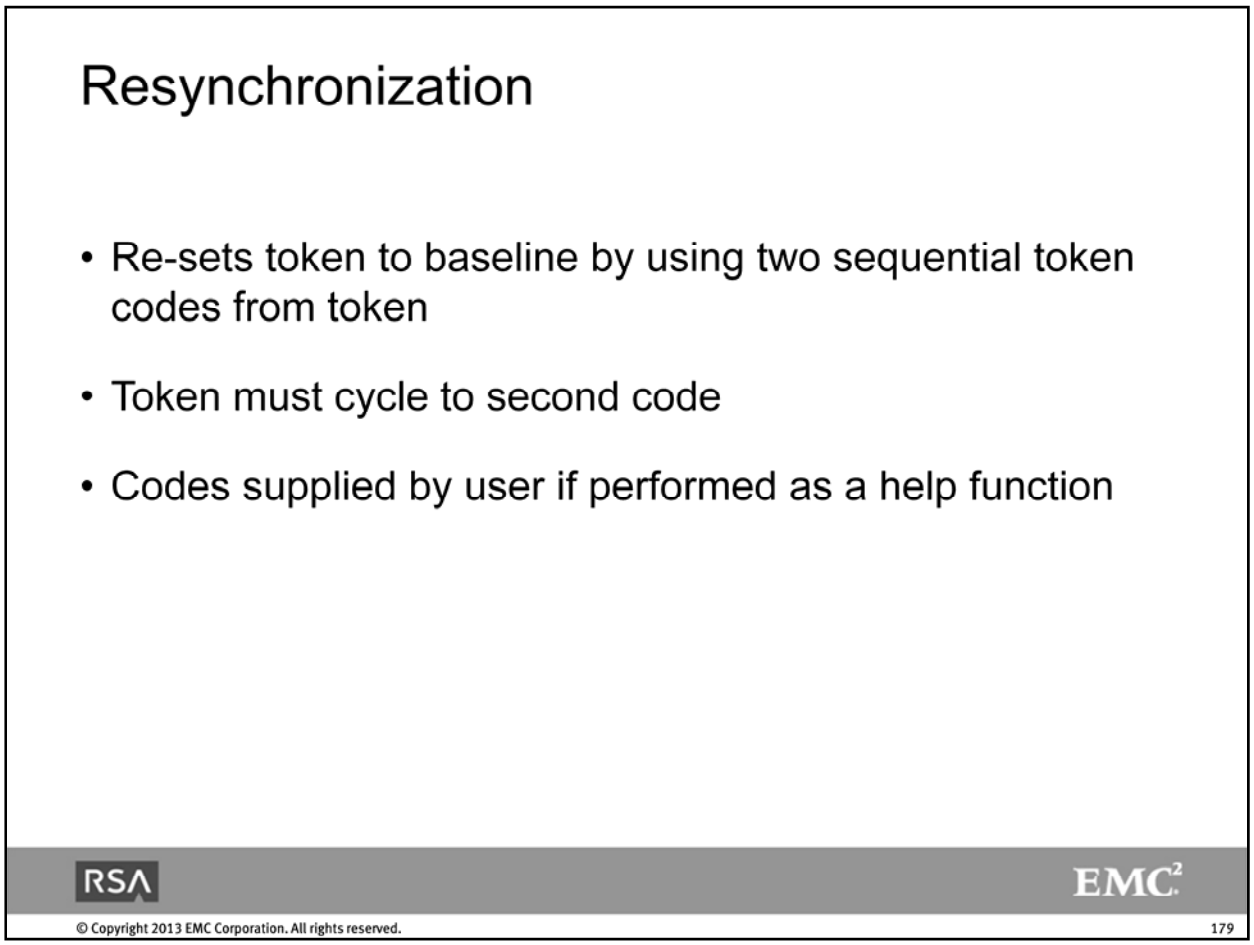
Tokens may have several states as described below:

- Unassigned - Indicates the token has not been assigned to a user and is available
- Assigned - Indicates the token is currently associated with a user account
- Disabled - Indicates the token record is active in the system but can not currently be used for authentication - even if the user provides a valid code from the token
- Deleted - Removes the token from the system and database

Disabled tokens can be re-enabled and assigned to other users; deleted tokens are unavailable in the system.

If a token is deleted from the system, the record may be recovered by re-importing the original token record but this operation should be done careful and thoughtfully because the original set of token records can potentially overwrite and result in unassigning other tokens from users.

Replaced by - indicates that a replacement token was assigned to a user account. Once the replacement token is used, it becomes active and the current active token becomes disabled.



Resynchronizing a token re-establishes the match between the expected code in Authentication Manager and the code displayed on the token.

Resynchronization typically is performed as a help desk function when a user is having trouble authenticating and no reason (other than incorrect token codes) for the problem can be determined.

Sometimes - especially with older or previously assigned tokens - an issuer will perform a resynchronization before assigning the token to a new user.

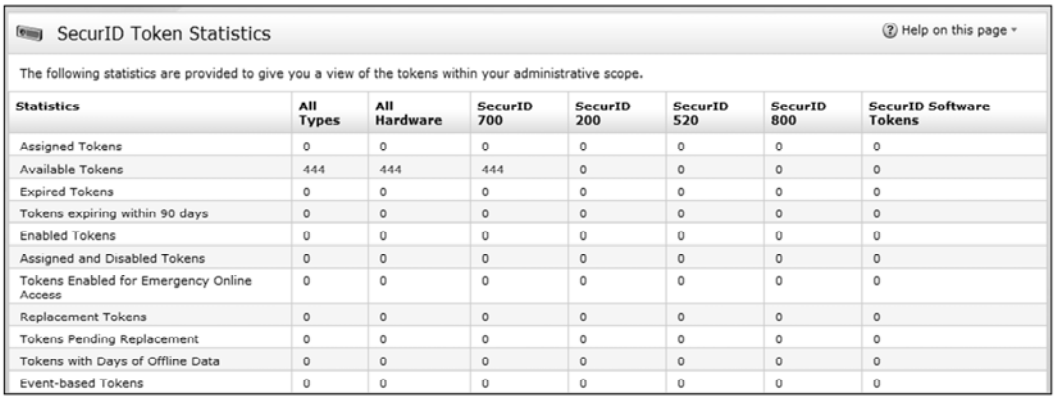
The resynchronization operation requires two sequential tokencodes from the token that the administrator enters into the resynchronization screen of the Security Console. In the case of a time-synchronous token, the codes can take up to two interval cycles to gather the sequential codes. in the case of event-synchronous tokens, the second code is obtained by pressing the sequence button of the token.

If resynchronization is accomplished over the phone with a user, the help desk administrator needs to communicate the procedure to the user and emphasize the accuracy of the code supplied by the user.

One the token is resynchronized, the baseline for that token is re-established and the token and authentication engine will supply and expect the correct token code at the next authentication.

Token Statistics

- Quick reporting tool to show token usage
- Shows only tokens within scope of admin running statistics
- Shows # of tokens, type, expiry summary, disabled, etc.



SecurID Token Statistics

Help on this page

The following statistics are provided to give you a view of the tokens within your administrative scope.

Statistics	All Types	All Hardware	SecurID 700	SecurID 200	SecurID 520	SecurID 800	SecurID Software Tokens
Assigned Tokens	0	0	0	0	0	0	0
Available Tokens	444	444	444	0	0	0	0
Expired Tokens	0	0	0	0	0	0	0
Tokens expiring within 90 days	0	0	0	0	0	0	0
Enabled Tokens	0	0	0	0	0	0	0
Assigned and Disabled Tokens	0	0	0	0	0	0	0
Tokens Enabled for Emergency Online Access	0	0	0	0	0	0	0
Replacement Tokens	0	0	0	0	0	0	0
Tokens Pending Replacement	0	0	0	0	0	0	0
Tokens with Days of Offline Data	0	0	0	0	0	0	0
Event-based Tokens	0	0	0	0	0	0	0

RSA EMC²

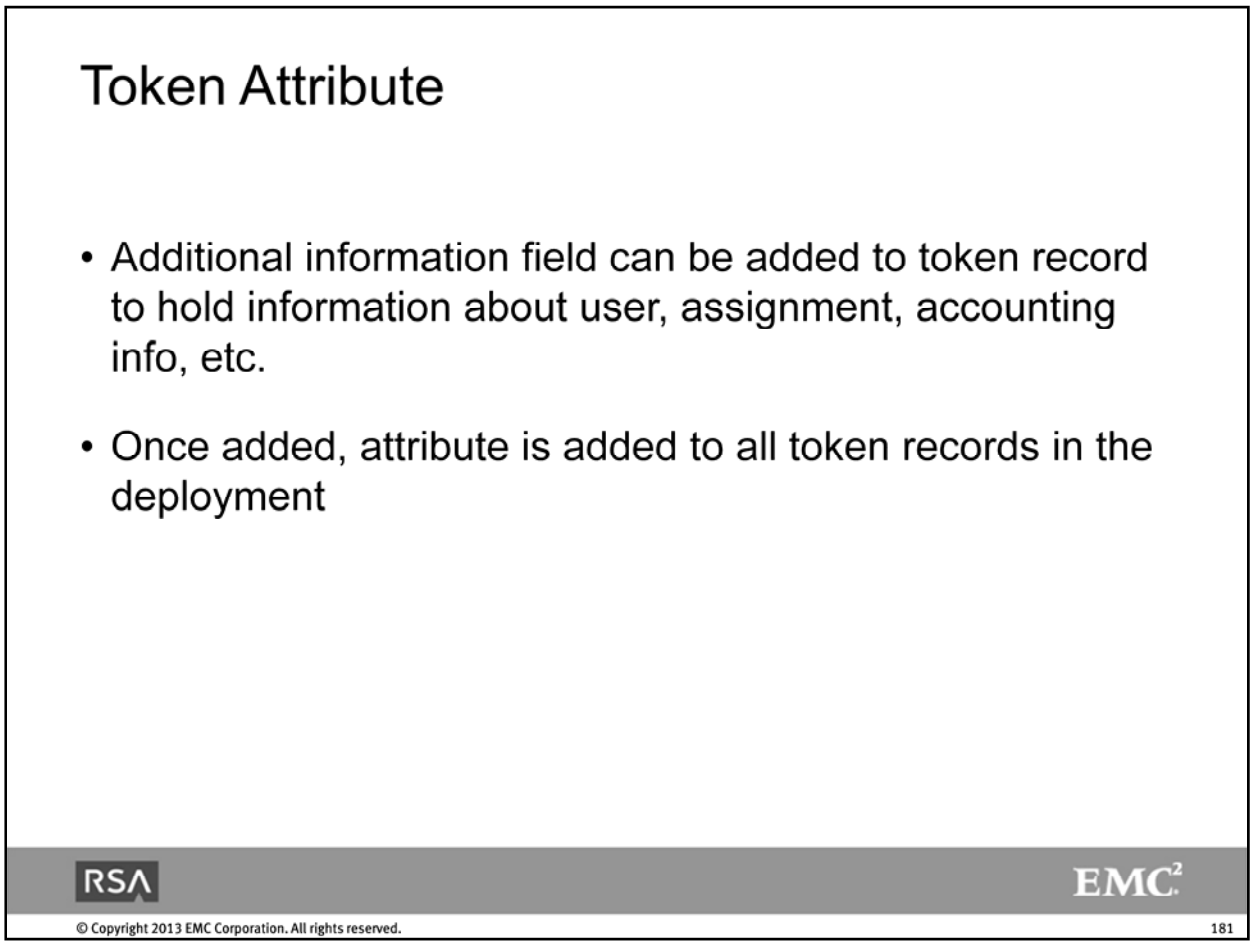
180

The Token Statistics table is a form of report that allows the administrator to view information on the tokens in the system.

The Token Statistics table shows only the tokens within the scope of the administrator running the report.

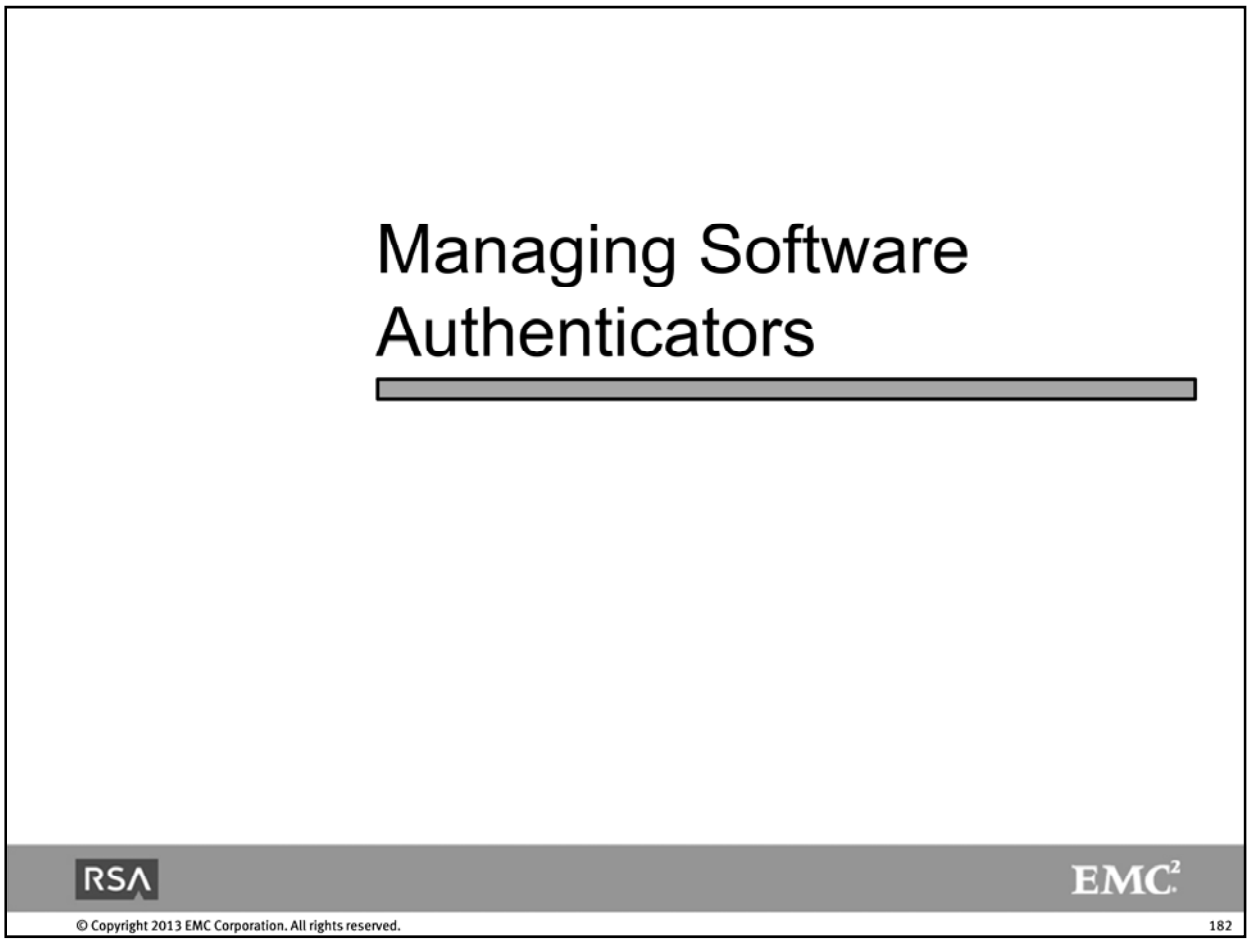
Token Statistics displays the number of tokens by type (key fob, PINPad, Software, etc.) that are:

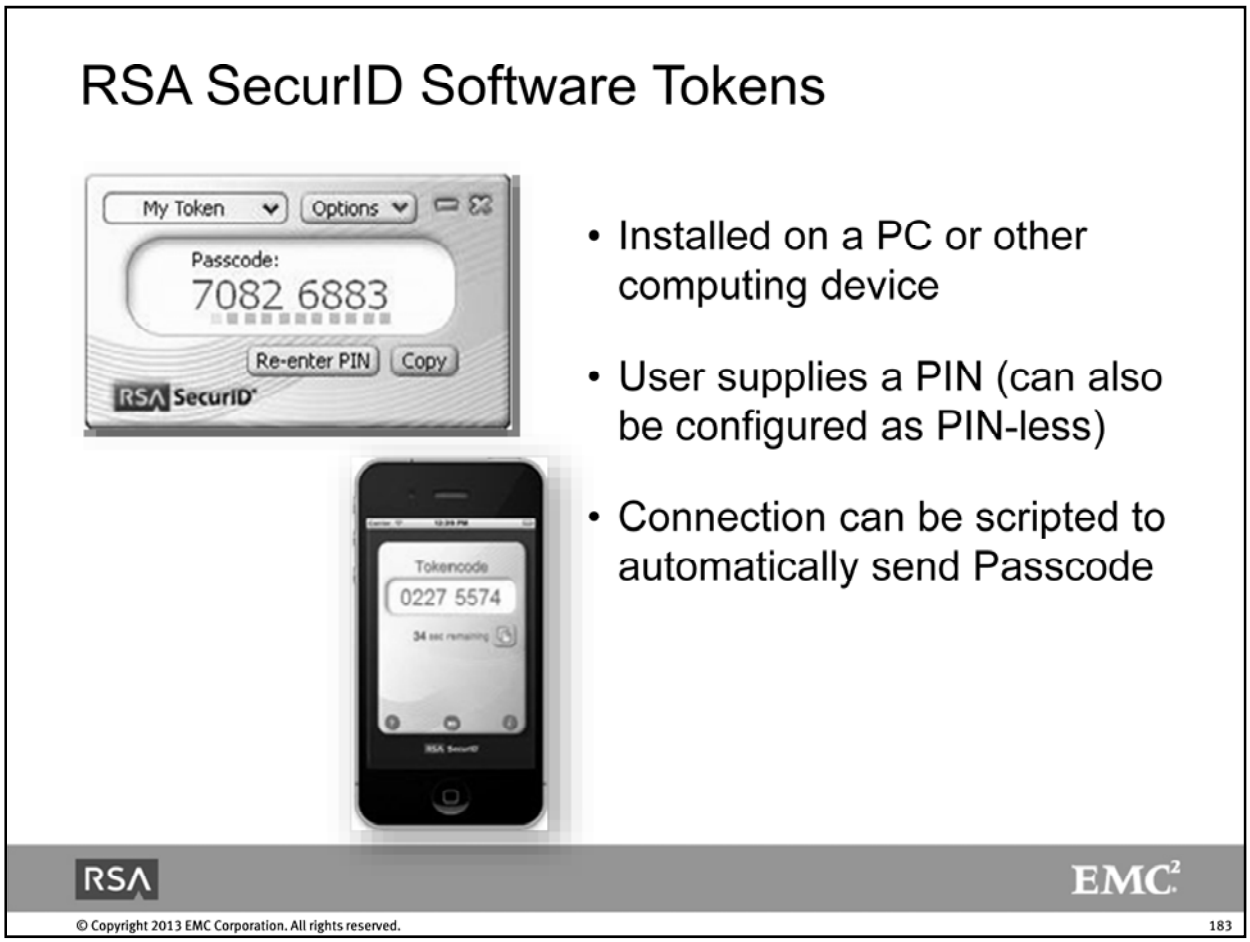
- Assigned
- Available
- Expired
- Expiring within 90 days
- Enabled
- Assigned and Disabled
- Enabled for Emergency Online Access
- Replacement Tokens
- Pending Replacement
- Tokens with Days of Offline Data
- Event-based Tokens



A Token Attribute is an additional field of information that can be used to hold information useful to administrators about a token. This field might be used to store additional information about the user, the token assignment, accounting information, or historical data.

For example, a token attribute useful for a Software Token might be the type of device on which the token is installed - such information might be helpful when troubleshooting a user authentication problem.


Managing Software Authenticators
RSA
EMC²
© Copyright 2013 EMC Corporation. All rights reserved.
182

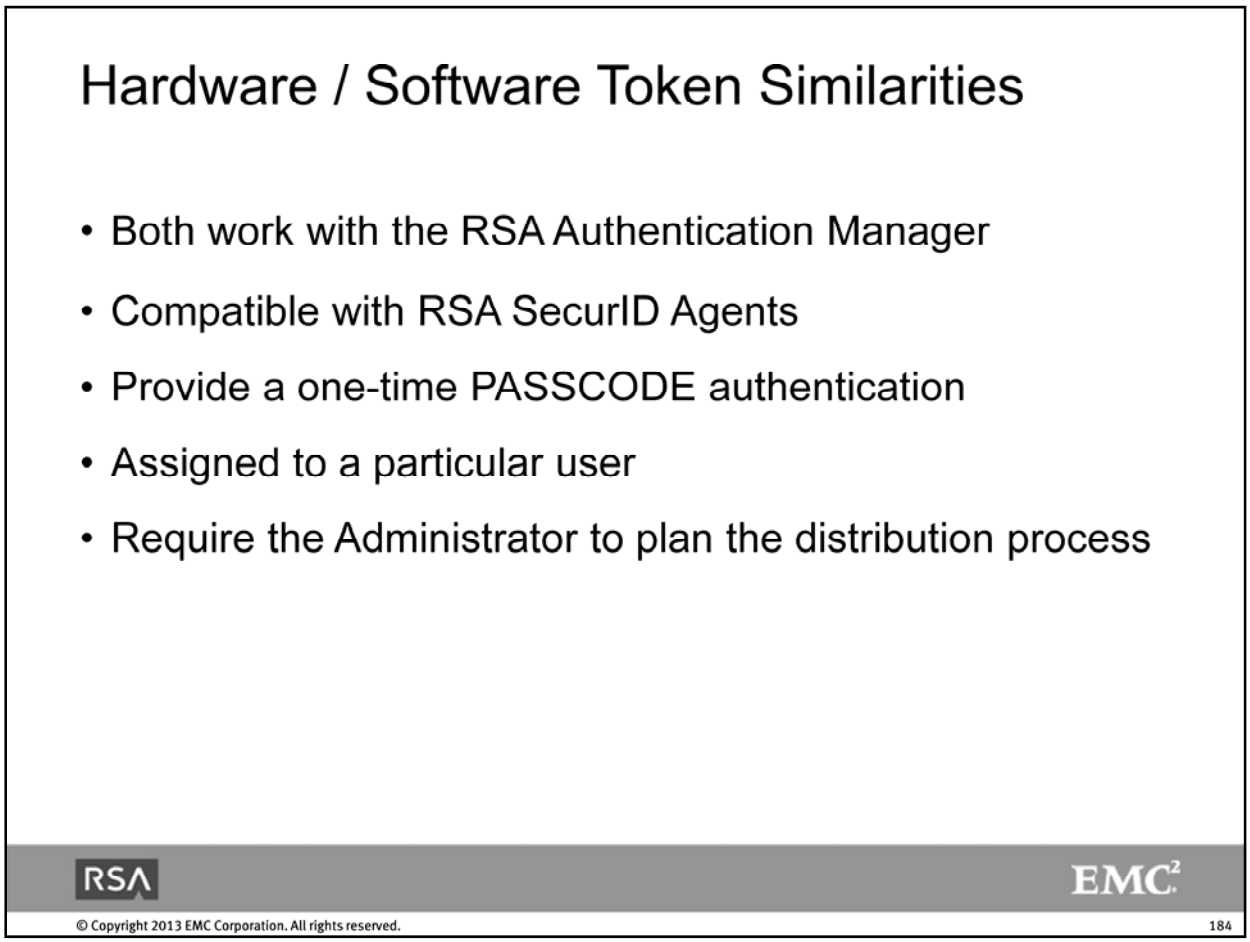


RSA SecurID Software Tokens are a software-based authenticator. RSA SecurID Software Tokens can be used in place of, or in addition to, an RSA SecurID hardware token to provide strong, two-factor authentication.

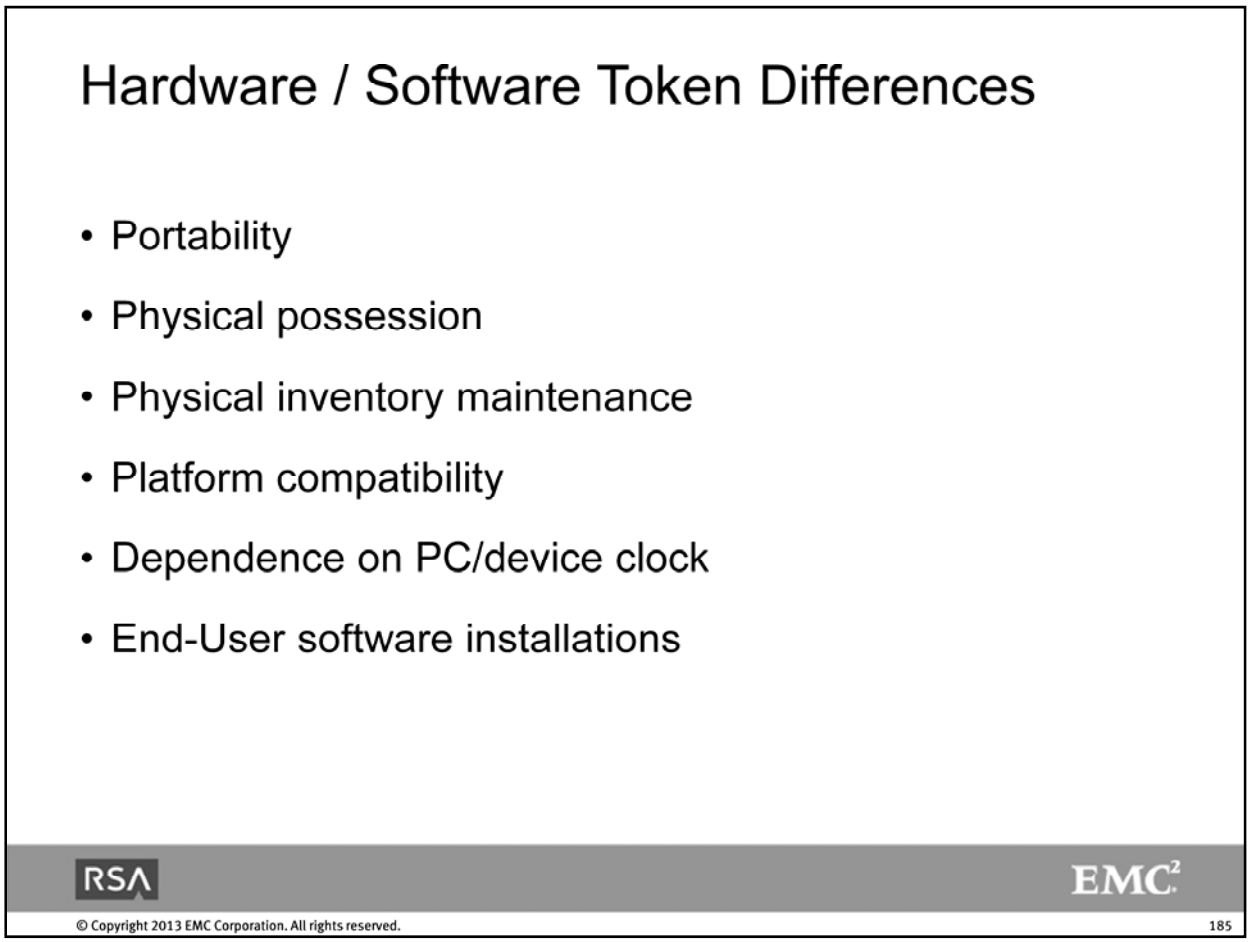
Since Software Token software is installed on a computer (typically, some type of portable device), the combination of the computing device and token software comprises an appliance similar to a hardware token – the combination, in effect, becomes a user's 'token'. RSA SecurID Software Tokens are generally used in remote access authentication situations.

Since the concept of two-factor authentication relies on "something you know" and "something you possess", possession of the computer or device containing the Software Token is critical to security. For example, if a user's laptop computer is on a desk – accessible to any passersby – authentication security relies on the strength of the user's PIN. Therefore, it is important for the user, you, as the security administrator, and your security policy to keep the physical security of user's computing device in mind.

RSA SecurID Software Tokens work in conjunction with RSA Authentication Manager and may be administered directly through the Security Console.



- Both hardware and software tokens work in conjunction with an RSA Authentication Manager.
- Both hardware and software tokens can be used with any Agent device compatible with an RSA Authentication Manager.
- Both hardware and software tokens provide a one-time PASSCODE authentication, which is superior to static passwords.
- Both hardware and software tokens are assigned to a particular user and user login in the RSA Authentication Manager database.
- Both hardware and software tokens will require the System Administrator to plan the distribution process carefully.



RSA SecurID hardware tokens are extremely portable, allowing the user to log in using any available device. A Software Token is loaded directly onto the user's machine; therefore it is only as portable as the device itself.

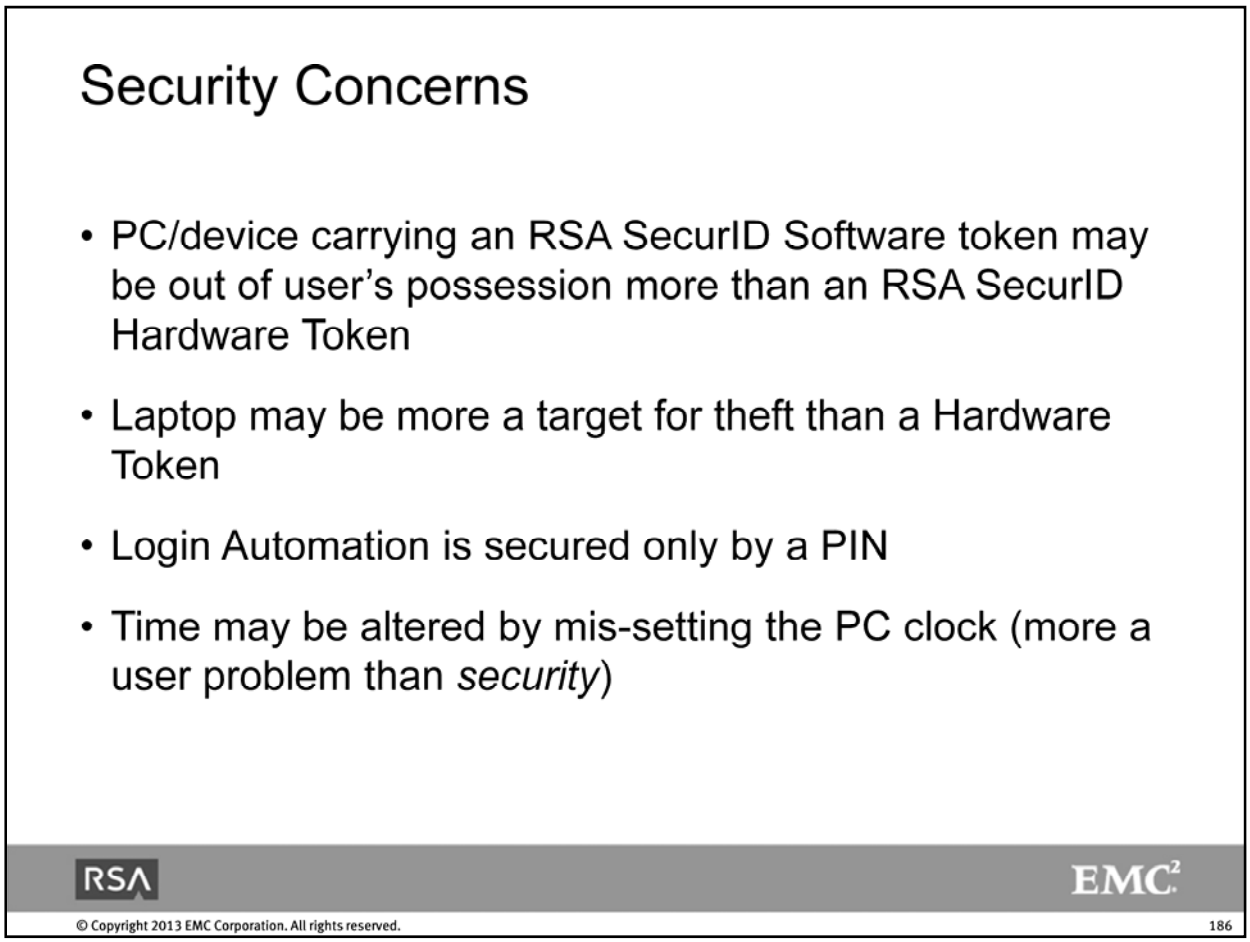
RSA SecurID hardware tokens should be carried by the end users, thus ensuring possession of a tokencode at all times. Since a Software Token is loaded onto the user's computing device, it is unattended when the device is unattended. When unattended, it is more vulnerable.

A hardware token’s life span, in part, is due to battery life. A Software Token does not have the same physical limitation but is hard-coded to a specific period when purchased.

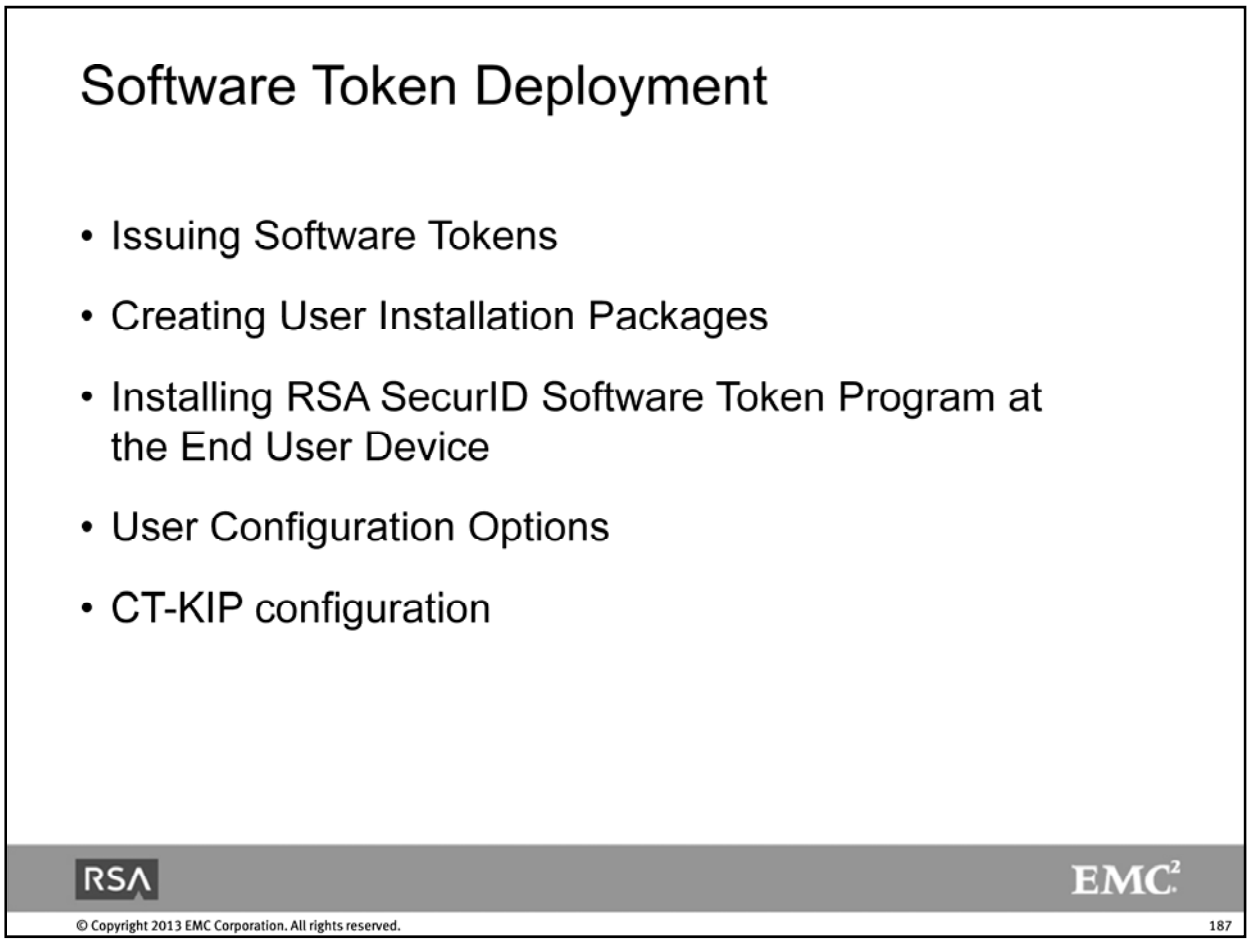
An RSA SecurID hardware token is independent of the operating system or version being used. Software Tokens must be used with compatible operating systems/versions and equipment.

RSA SecurID Software Tokens do not require a physical inventory.

A Software Token is dependent on the clock of the host device. The time setting, time zone, etc. need to be accurate. Hardware tokens have their own internal clock – independent of the Server or Agent computer's clock.



The other security concerns which many people have lie in issues such as possession or theft of a laptop computer hosting a Software Token. Such issues vary with the application, level of protection required, and Security Policy. The RSA SecurID Software Tokens fits well in many environments, some may be better suited to hardware tokens. Each administrator and organization must make this determination based on their particular needs.



There are two or three elements involved in supplying an RSA SecurID Software Token to an end user:

- The Software Token record – this is the Software Token seed information – issued through the Authentication Manager Security Console – that will allow the Software Token software to correctly produce tokencodes.
- The supporting Software Token software – this software provides the interface (and display) for the software token tokencodes as well as the environment for supporting login automation, if used (see below).
- (Optionally) A customized Login Automation script – this is a login script used with the Login Automation program that can be distributed to the end user in conjunction with the Software Token. Login Automation allows an RSA SecurID passcode to be submitted automatically as part of a connection script.

Exercise: Managing Authenticators

- Assign various authenticators to users in your system
  - Hardware and Software tokens
  - Distribute Software Token
- Test Authentication

RSA EMC²

188

Managing Risk-Based Authentication

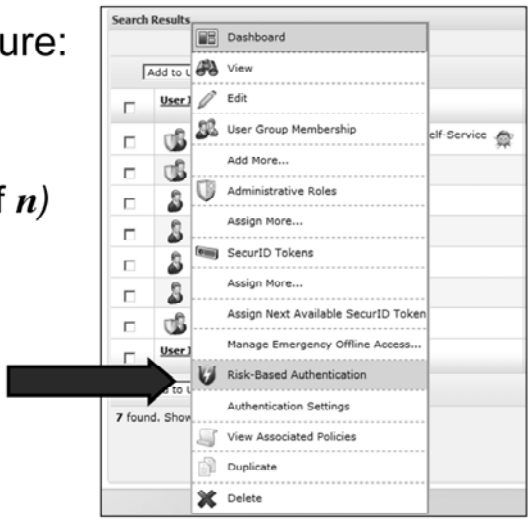
Unit 11

RSA EMC²

189

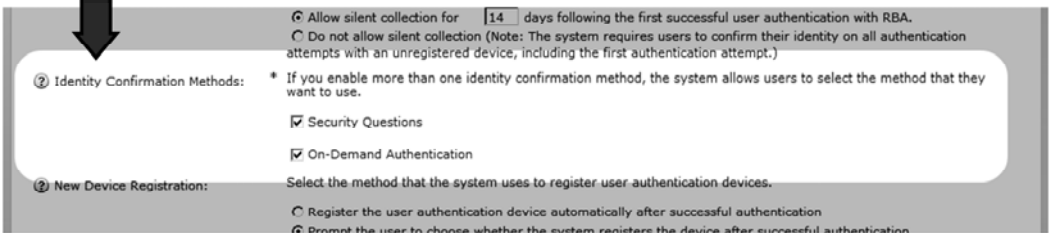
Enabling Users for Risk-Based Authentication

- To enable RBA, you first configure:
  - On-Demand Authentication parameters (if used)
  - Security Question policy (***m*** of ***n)***
  - RBA policy
- Risk-Based enablement is accessed through User screen (right-click pop-up)



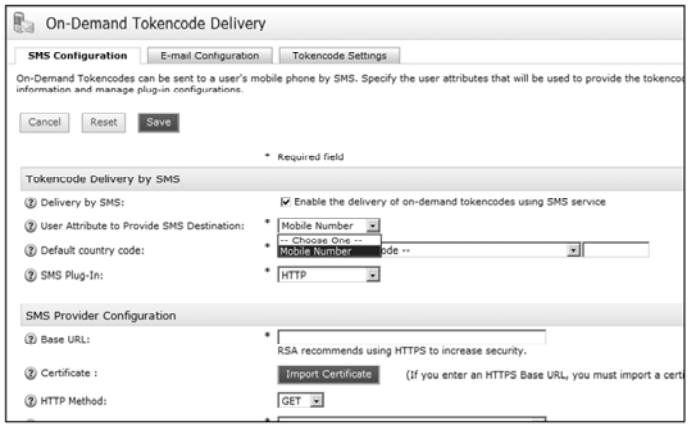
Enabling Users for Risk-Based Authentication *(cont'd)*

- User is enabled via checkbox
- Identity Confirmation method(s) depends on RBA policy settings



Setting up On-Demand Authentication

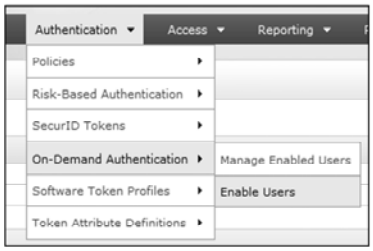
- On-Demand Authentication set up through **Setup > System Settings** screen: **On-Demand Tokencode Delivery** link
- Defines:
  - User attribute for SMS delivery
  - SMS Provider info (with HTTPS and Proxy info, if needed)



- Can also be set up for e-mail delivery of OD tokencode

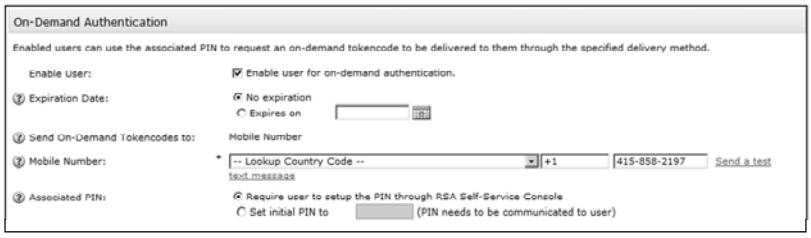
Enabling On-Demand Authentication

- Users are enabled for ODA via **Authentication > On-Demand Authentication > Enable Users**



- User is enabled from right-click pop-up or with checkbox and **Enable for ODA** control

- Other options allow setting expiry and PIN options



On-Demand / Risk-Based Configuration (Self-Service Console)

- When user logs on to Self-Service Console, ODA / RBA configuration status is displayed
- User can create ODA PIN if PIN was not established by an Administrator
- RBA status provides info on registered devices

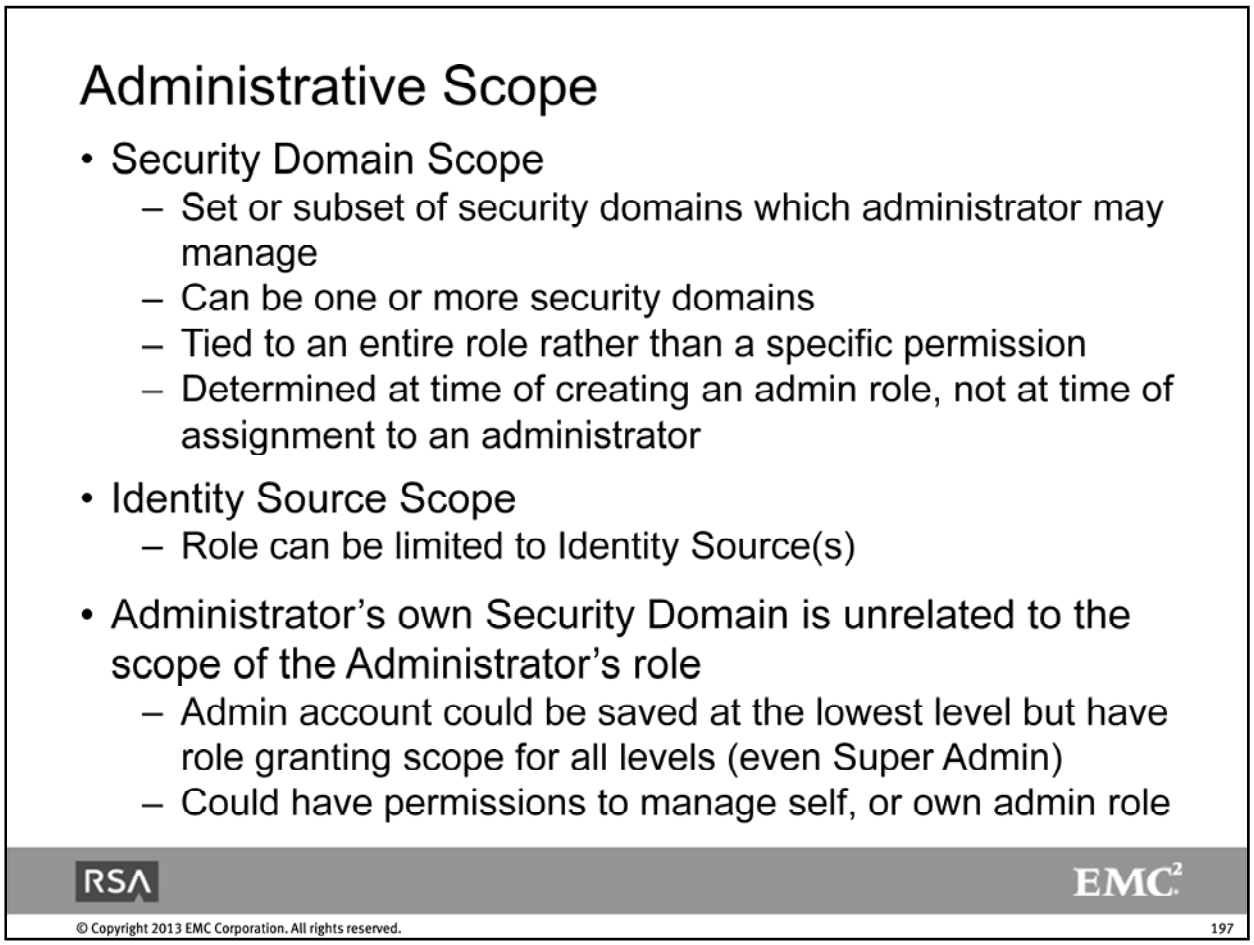
Exercise: Configuring Risk-Based Authentication

- Configure Risk-Based Authentication settings for system and users

RSA EMC²

Delegated Administration

Unit 12



Administrative Scope defines which objects in the system an administrator may operate upon. The scope is defined by Security Domain and/or Identity Source

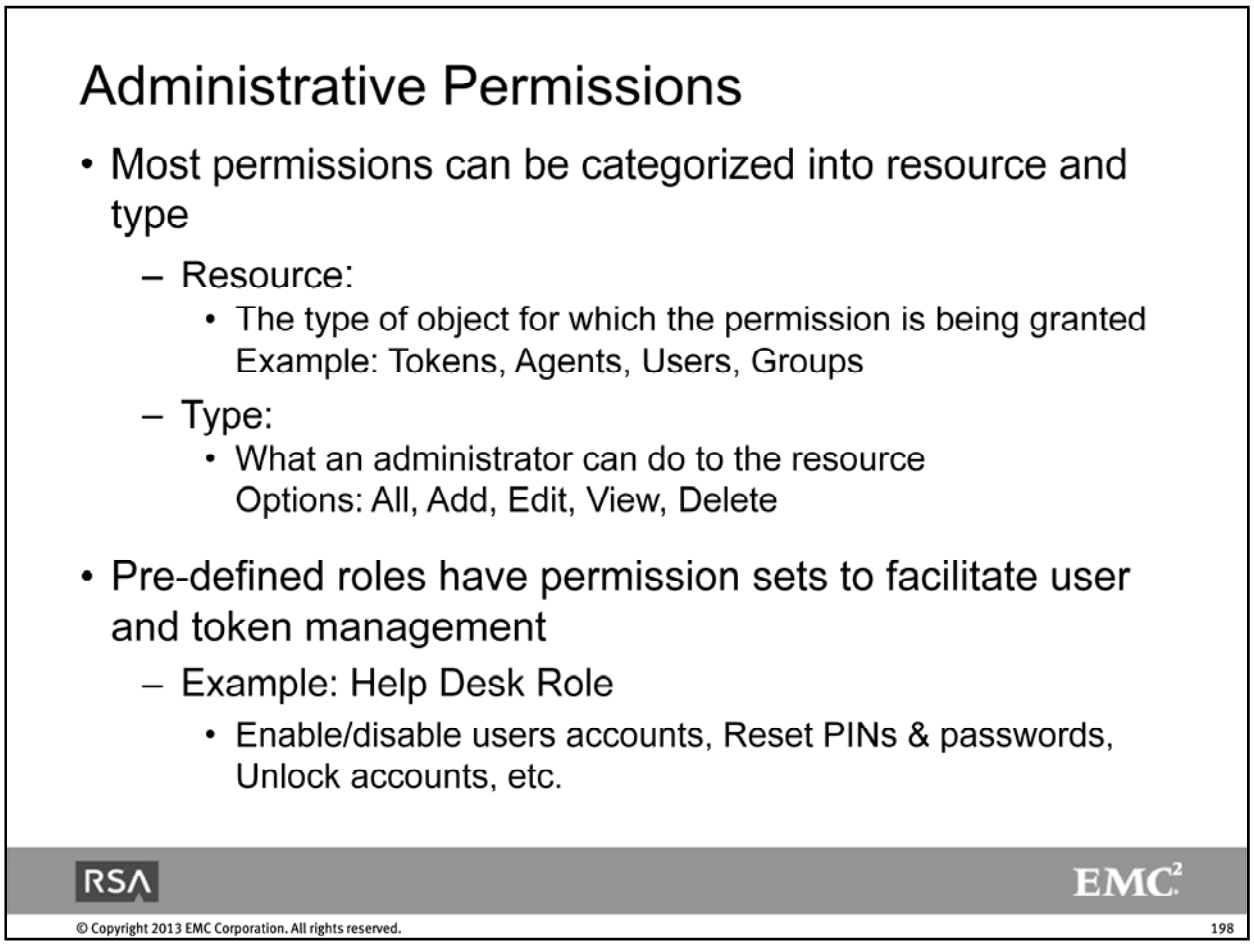
Security Domain Scope: This is the set or subset of security domains that an administrator is allowed to manage. One or more security domains can be specified in the scope.

The scope applies to the entire administrative role rather than to a specific permission and is determined at time of creating an administrative role, not at time that the role is assigned to an administrator.

Identity Source Scope: An administrative role can be limited in scope to an identity source. This applies only to Realms that are linked to two or more identity sources.

The security domain in which the administrator resides is unrelated to the role's scope. An administrative account could be saved at the lowest level security domain but have a role granting scope for all levels (even the Super Admin account).

An administrator can even have permissions to manage him/herself, or his/her own administrative role.



Most permissions can be broken down into resource and type.

Resource: This indicates the type of object for which the permission is being granted. For example: tokens, agents, users, or groups.

Type: This indicates what an administrator can do to the resource. The options for permission type are: Add, Edit, View, or Delete.

There are also special action permissions such as “Assign token to user” or “Enable/Disable Users”.

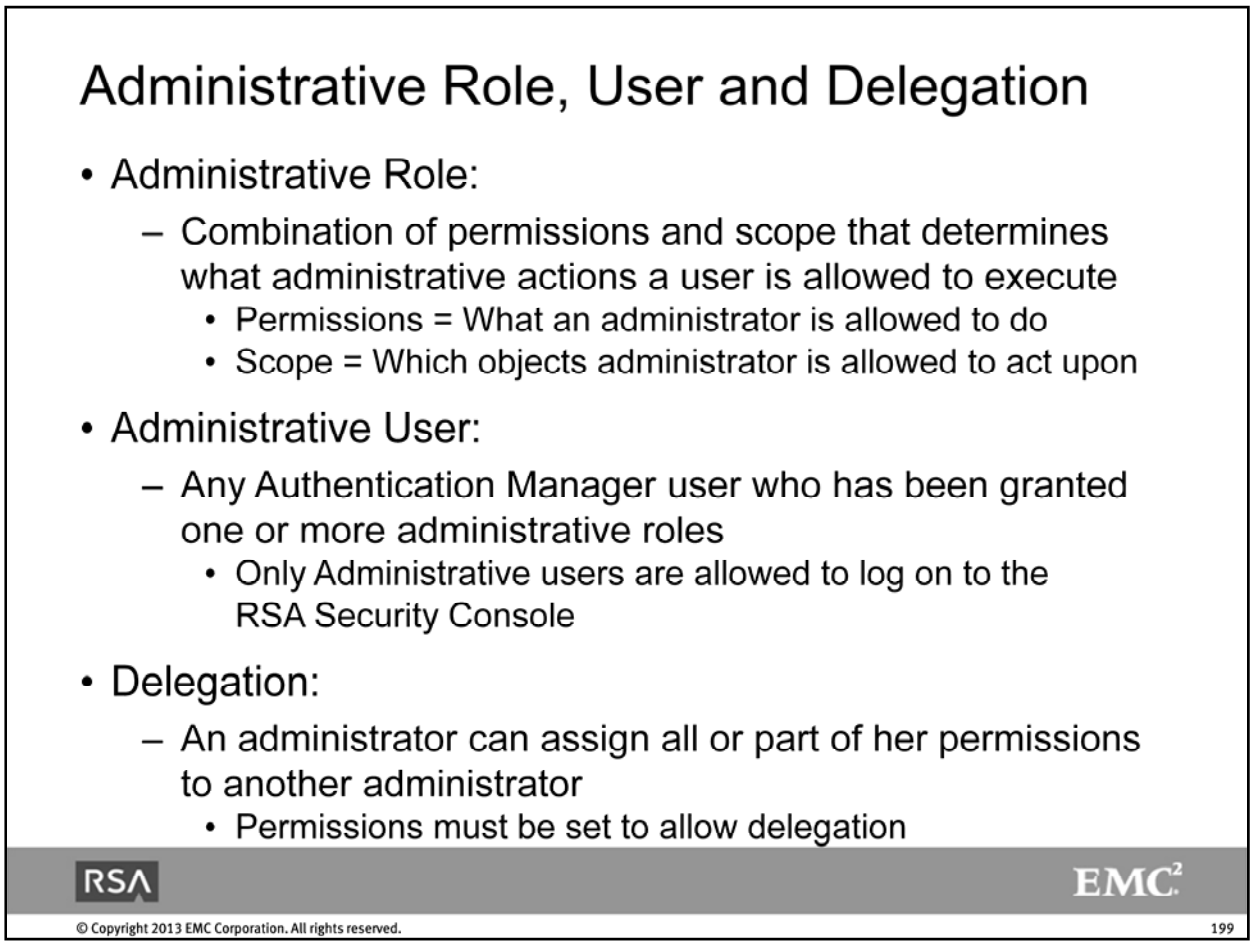
The pre-defined administrative roles in the system have permission sets to facilitate user and token management. For example, the Help Desk Role contains permissions to enable/disable accounts, reset passwords, clear PINs, require PIN change, etc.

These permission sets are intended to provide the most common actions for an administrator of each type. Any permission set can be customized to better serve a particular organization by creating a new administrative role.

Certain permissions are only available if a role has scope over an entire realm. For example: Permissions for data associated with realm rather than a specific security domain (policies, trusted realms, console display options) e.g., the “Realm Administrator” role

There is no special casing of user permissions for administrators - any administrator can update or delete another administrator if granted the appropriate permissions and scope - even Super Admin.

To avoid potential problems, it is better not to put all administrators in the same security domain. In this way, scoping control can be applied to prevent too much control for any single administrator over all others.



Administrative Role

An Authentication Manager Administrative Role is comprised of a combination of permissions to perform certain actions and the scope within which those actions can be taken.

The permission set determines what an administrator is allowed to do.

The scope defines which objects - in which Security Domains or in which Identity Sources - the administrator is allowed to act upon.

Administrative User

Any Authentication Manager user who has been granted one or more administrative roles becomes an "administrative user" in Authentication Manager.

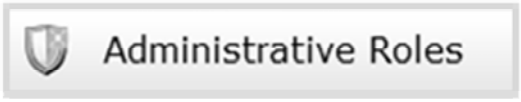
Only administrative users are allowed to log on to the RSA Security Console and perform operations tasks.

Administrative Delegation

An administrator can assign all or part of his or her permissions to another administrator. For this to be possible, the permissions must be specified as delegatable.

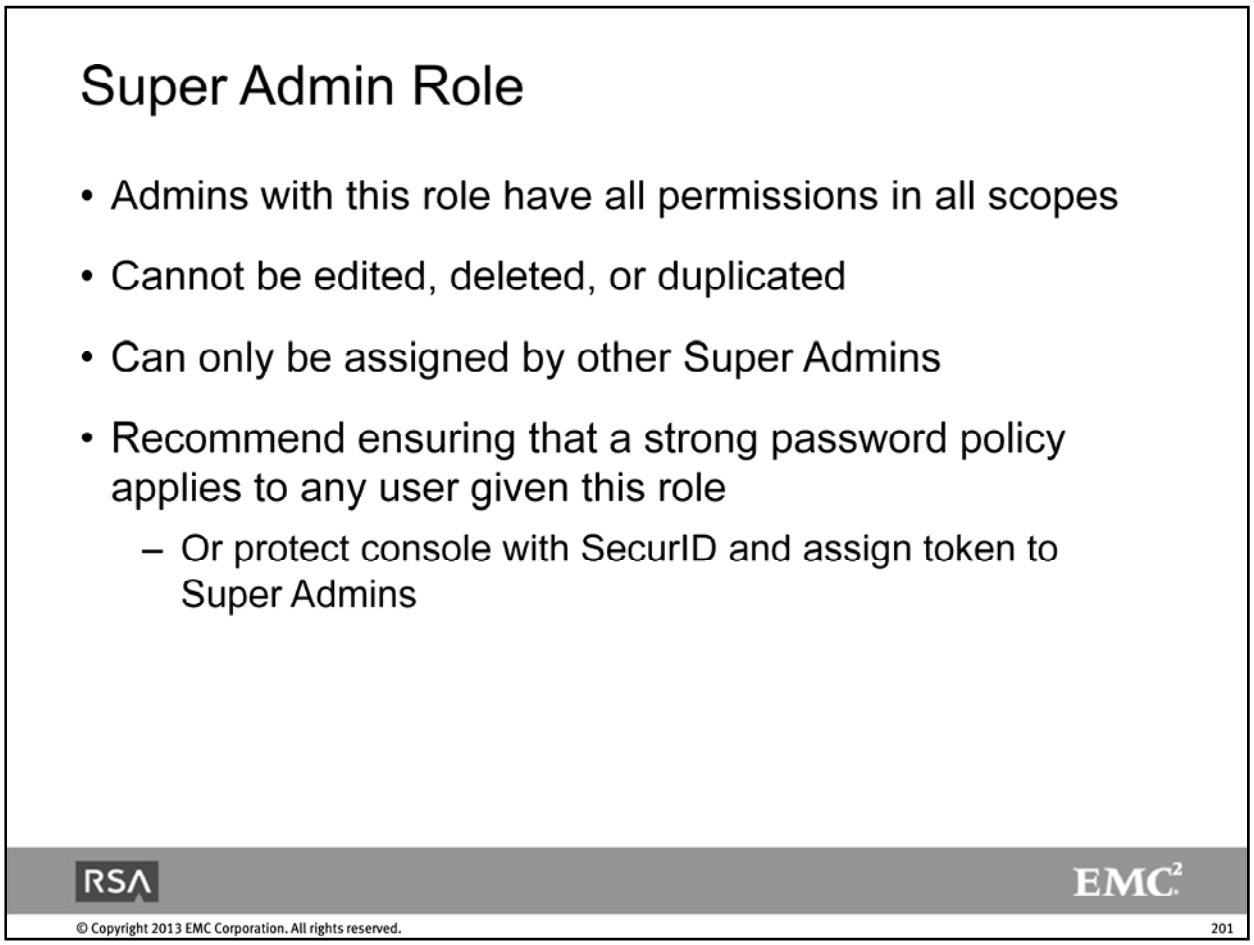
Levels of delegation can technically continue indefinitely (n-level delegation). The practical limit for a given organization will depend on the organization's size, the number of administrators tasked with working with Authentication Manager, security policy, and amount of control desired for each administrative user.

Pre-defined Roles



- Super Admin
- Security Domain Administrator
- User Administrator
- Token Administrator
- Help Desk Administrator
- Privileged Help Desk Administrator
- Agent Administrator
- Request Approver
- Token Distributor
- 13 roles in total

RSA EMC²



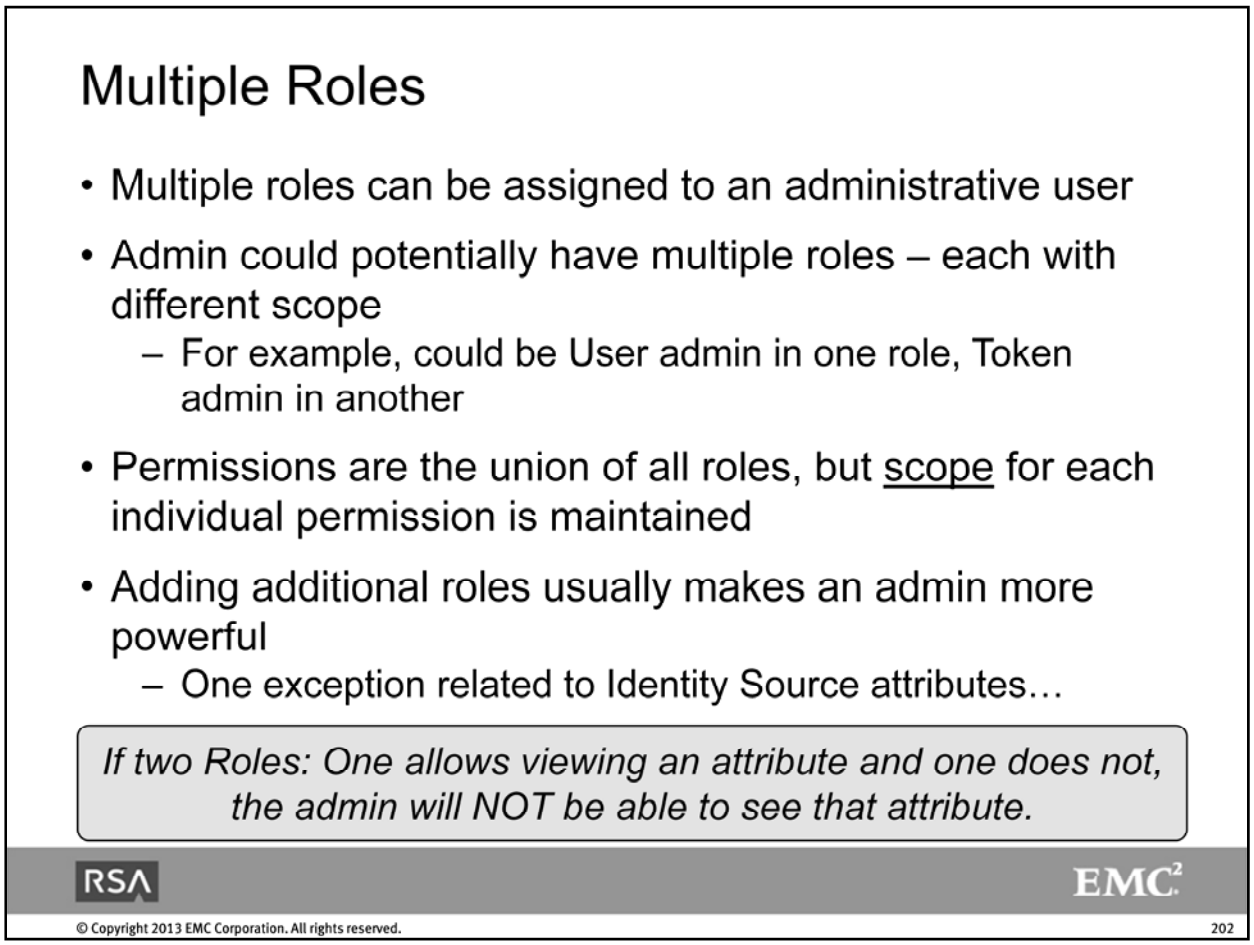
The Super Admin role is the role with the greatest power and control in the system. At installation, the default Super Admin role is assigned to the initial administrator account.

Administrators with this role have all permissions in all scopes and this is the only role which crosses realms.

The Super Admin role cannot be edited, deleted, or duplicated. It can be assigned to create other administrators (for example, a 'Backup Super Admin') but only assigned by a user that themselves have the Super Admin role.

The role can be viewed by non-superadmins who have view role permission over whole realm but not acted upon.

It is recommended that a strong password policy be applied to any user given the Super Admin role or protect access to the Security Console with SecurID and assign Super Admins an authenticator.



As many roles as desired can be assigned to an administrative user. This means that administrators can have multiple roles, each with a potentially different scope. For example, an administrator could be User administrator in one scope; a token administrator in another.

Permissions are the union of all roles but the scope for each individual permission is maintained.

Adding additional roles always makes an admin more powerful. That is, you can not add an additional role to a user that will reduce that administrator's scope or permission set.

There is one exception to this rule related to identity source attributes: If an administrator is assigned two roles, one of which allows viewing an attribute and one that doesn't, the administrator will not be able to see that attribute.

Administrative Role Pitfalls

- Associative permission
  - Example: Admin can Edit Users, can Edit Tokens but may not have permission to assign a token to a user
- Multiple Roles
  - Example: Admin can Edit Users & Tokens and Assign tokens in one Security Domain but not another

Field Sales Admin Role
- View/Edit users, View/Edit tokens, Assign Tokens
- For "Inside Sales" Security Domain

Sales Operations Admin Role
- View users, View tokens
- For "Sales Ops" Security Domain

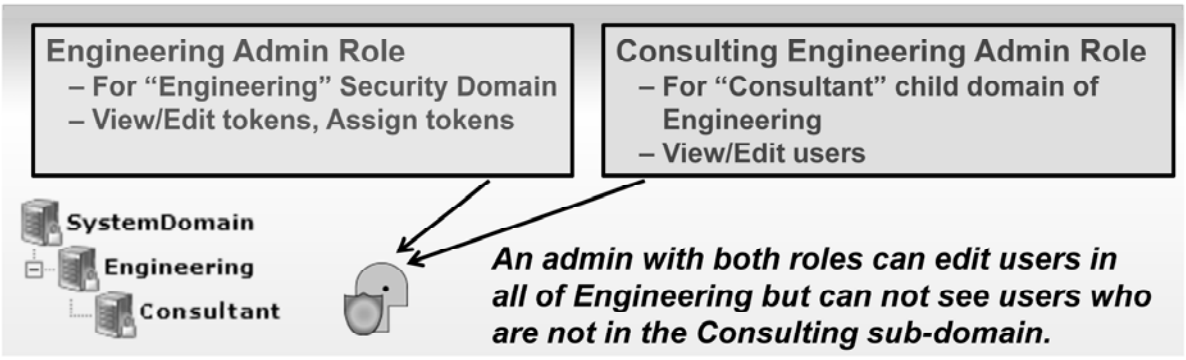
An admin with both roles can see users and tokens in both domains but can not assign tokens to Sales Ops users.

RSA EMC²

203

Administrative Role Pitfalls *(cont'd)*

- Overlapping Roles
  - Potentially confusing behavior if an admin has multiple roles with overlapping scope
  - Could potentially see objects and have access to an operation (assign token to user, for example) but not be able to execute the operation
  - Recommendation: Don't set up overlapping admin roles



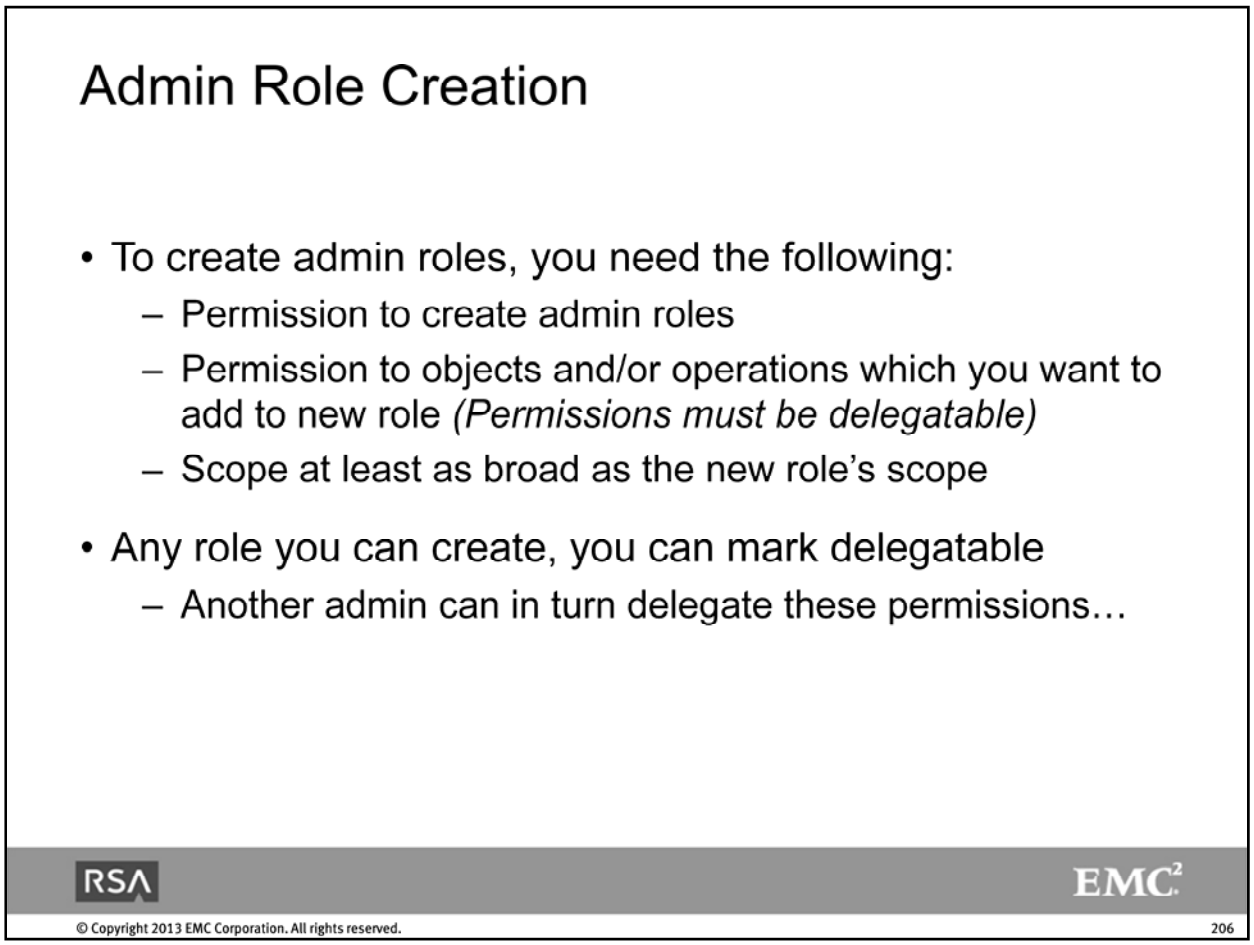
RSA EMC²

204

Administrative Role Pitfalls *(cont'd)*

- User Groups/Admin Roles Interaction
  - In certain situations, admins might affect users they cannot view
    - User groups may contain users from different Security Domains
    - Admin role may allow seeing some users in group but not others (see one Security Domain but not another)
    - Could grant user access to restricted agents without knowing it
  - Recommendation: do not set up domain model this way

RSA EMC²



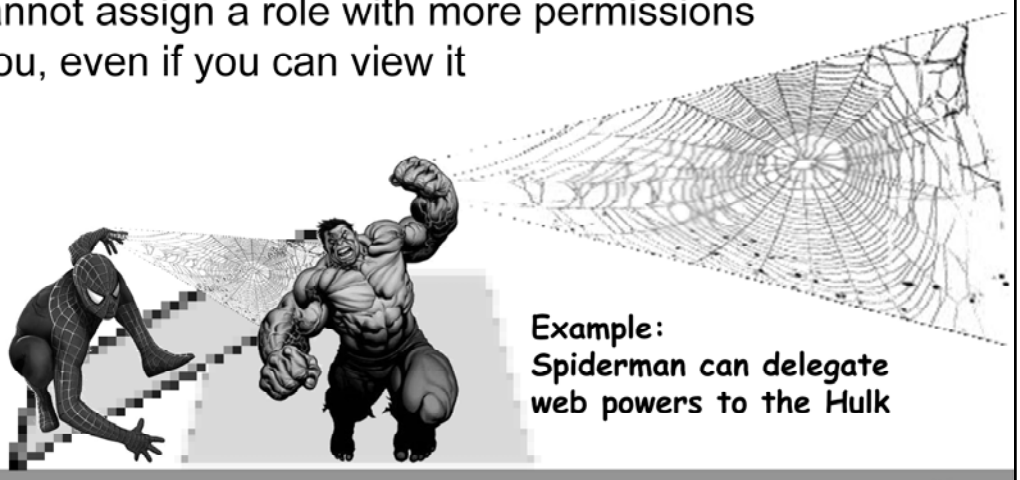
To create administrative roles, you need the following:

- Permission to create administrative roles
- Permission(s) which you want to add to new role (with scope at least as broad as the new role's scope)
- Permissions that you want to add to the new role must be delegatable

Any role that you can create, you can mark as 'delegatable' - another administrator can then, in turn, delegate these permissions.

Admin Role Delegation

- You can delegate any role which you are able to create (i.e. roles with power equal to or less than your own)
- You can also assign such roles even if they are not marked delegatable
- You cannot assign a role with more permissions than you, even if you can view it



Example:
Spiderman can delegate web powers to the Hulk

RSA EMC²

207

Exercise: Creating Administrators

- Define administrative users for your Authentication Manager deployment

RSA EMC²

(Optional) Exercise: Experiment with Delegation, Scope, and Role

- Set up scenarios where administrative accounts have various roles and responsibilities for multiple domains

RSA EMC²

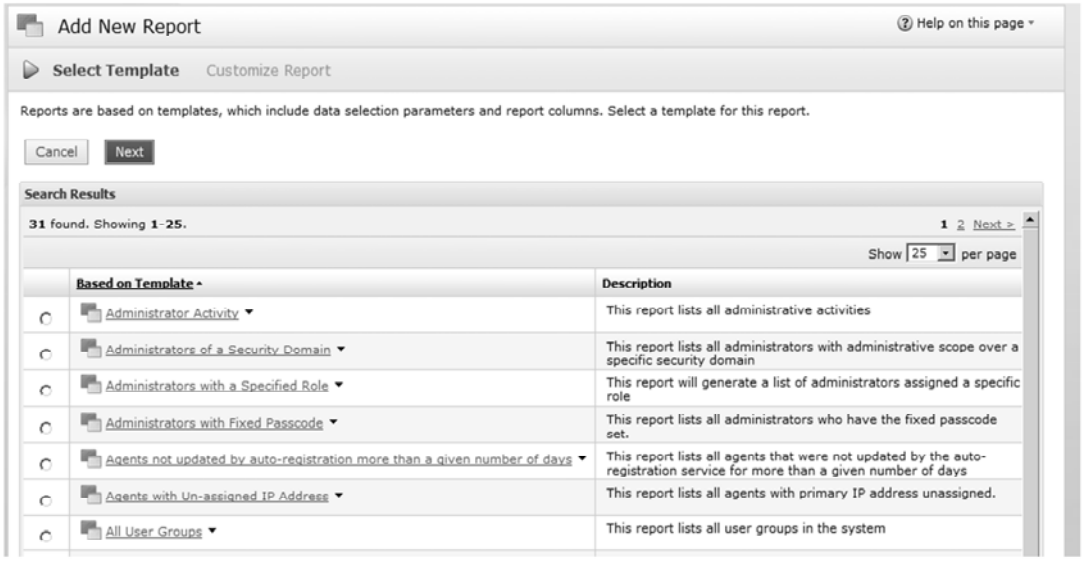
209

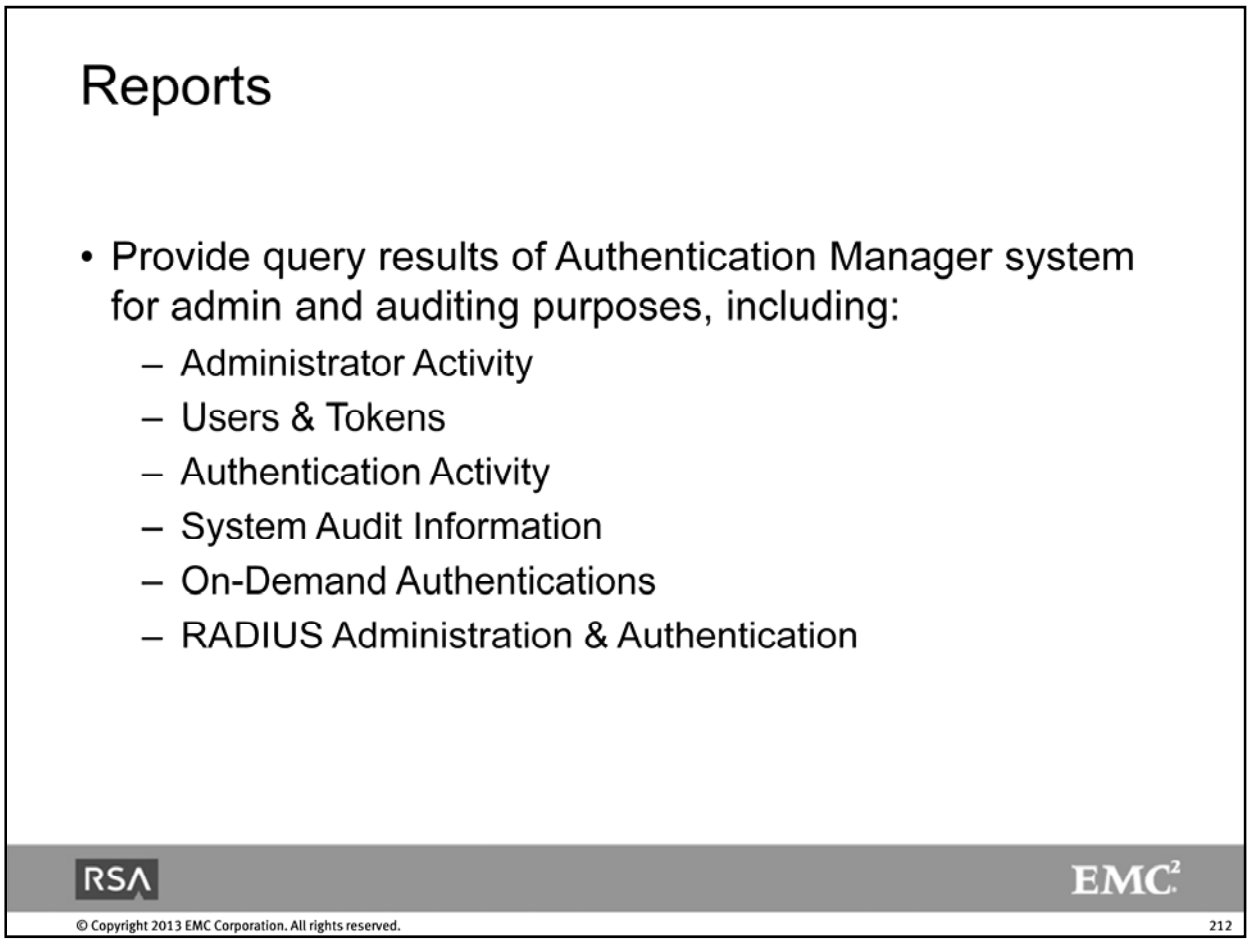
Reports and Logs

Unit 13

Security Console

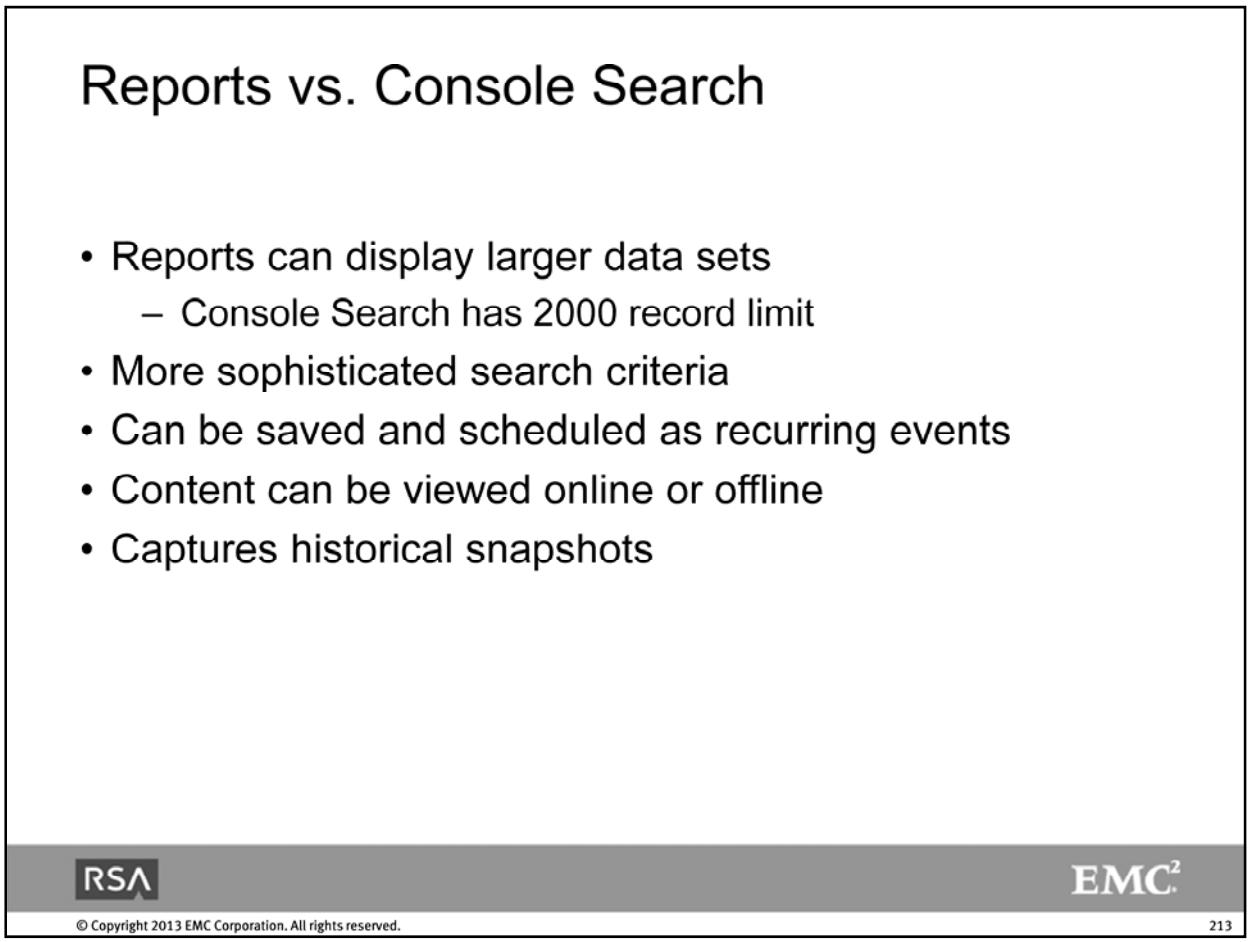
- Access Reporting via **Reporting > Reports** menu path





The Authentication Manager reports function provides query results of the system for administrative and auditing purposes. Reporting provides information about:

- Administrator Activity
- Users & Tokens
- Authentication Activity
- System Audit Information
- On-Demand Authentications
- RADIUS Administration and Authentication



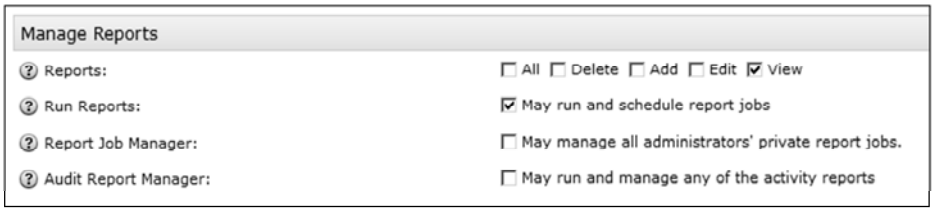
Compared to the console 'Search' capability, reporting typically provides larger data sets (the console search has a 2000 record limit) and allows more sophisticated search criteria.

Report criteria can be saved and can be scheduled and run on a recurring basis.

Report output is saved to a file and can be viewed both online and offline.

Administrative Privileges

- Security Console access for admin requires user id / password
- Admin role with Reports access allows:
  - Delete, Add, Edit, and View (template/query)
  - Run & Schedule (report jobs)
- Admin role with the Domain access:
  - View Domains
- Other permissions/Roles can be report-specific



RSA EMC²

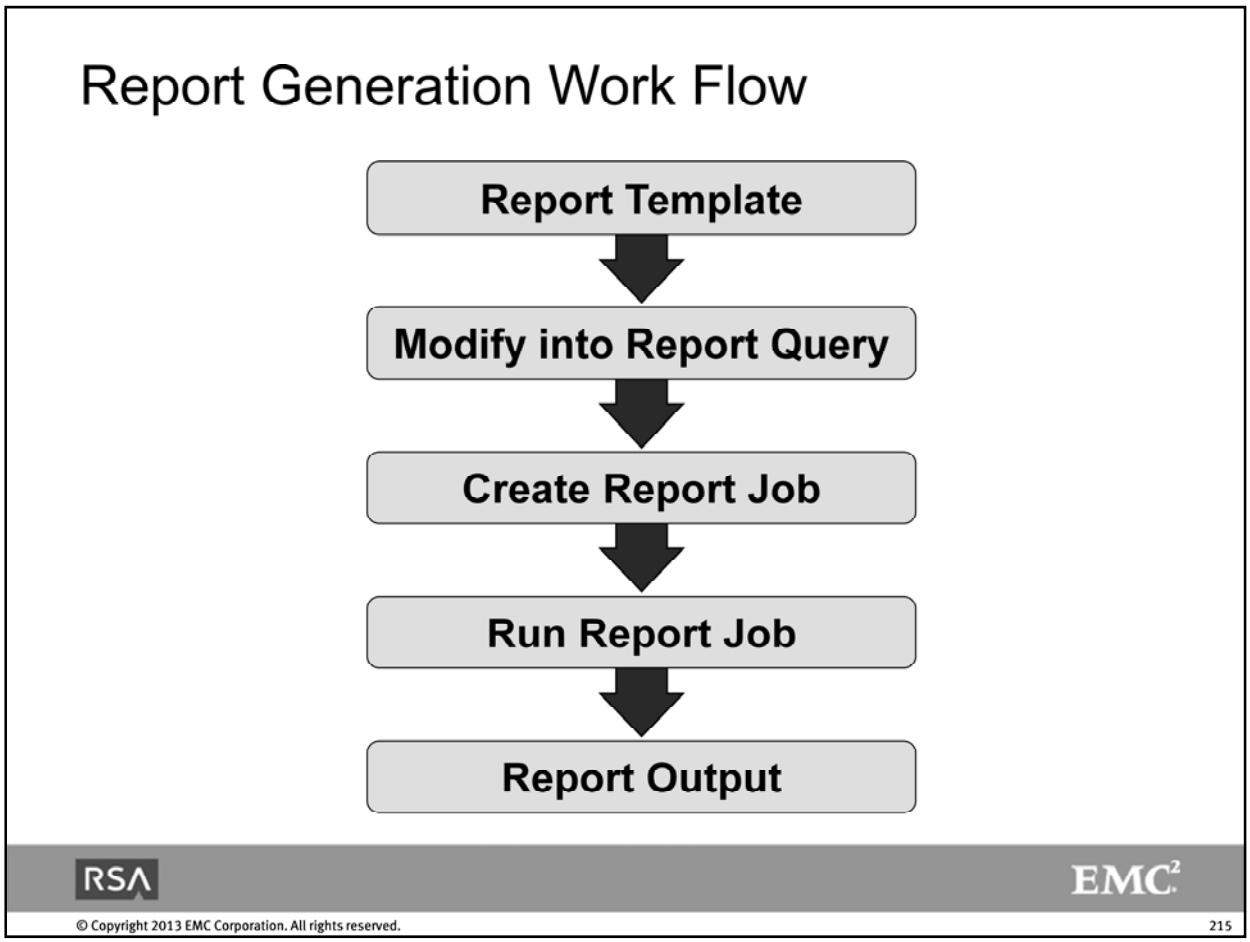
214

Access to the reporting function through the Security Console requires administrative authentication (at minimum, a userID and password).

An administrative role with Reports access allows the ability to Add, Edit, View, and Delete report queries and View report templates. This role can also Run and Schedule report jobs.

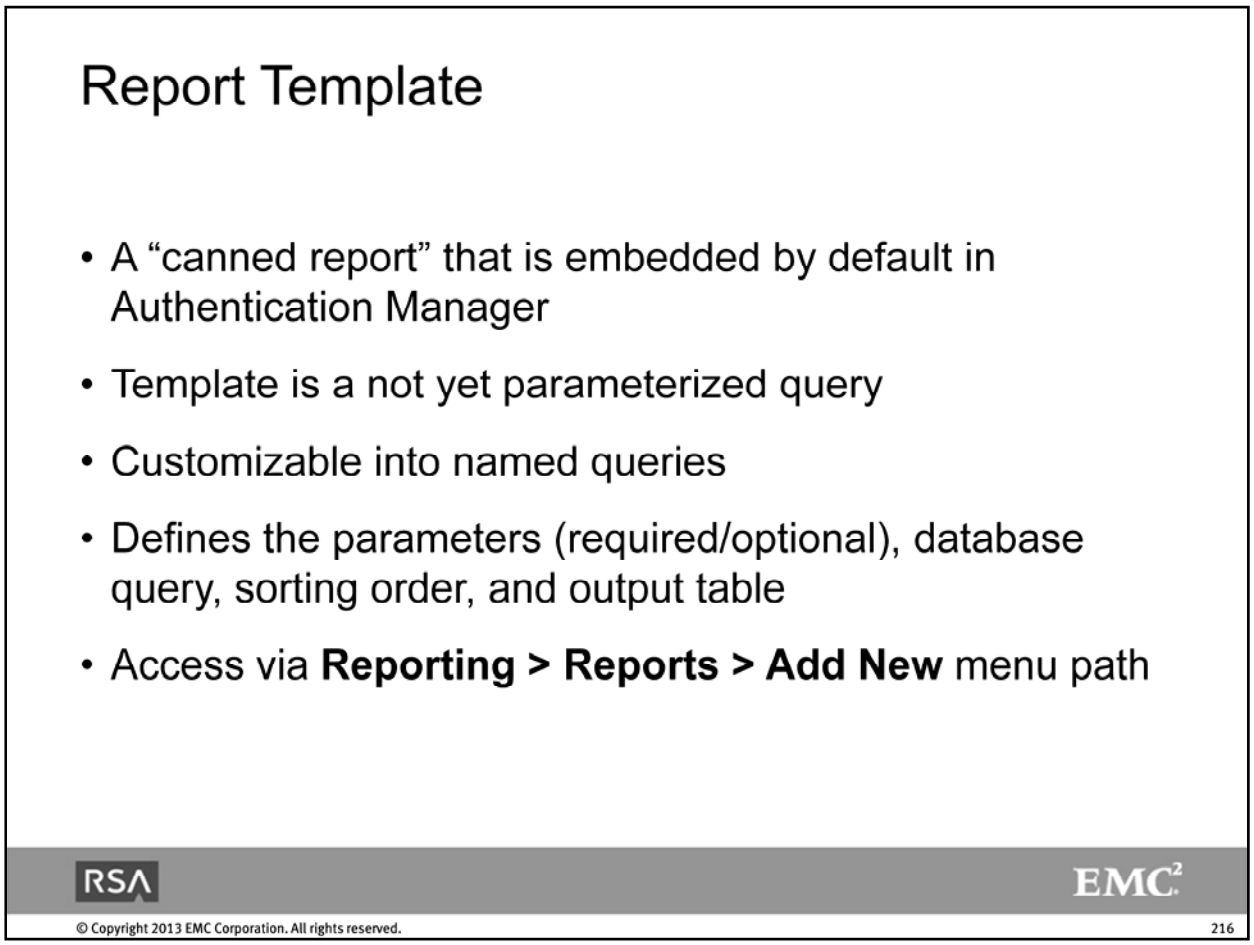
An administrative role with domain access can also view Security Domains.

Other permissions and roles assigned to an administrator can be specific to particular reports - each report can be set to allow permissions independently.



The workflow in the report generation process begins with a report Template. The template is then modified and customized to create a Query.

A Report Job is created to include the Query and scheduled to run on a one-time or on a recurring basis. The report then generates an Output to the database that can be viewed in a browser or downloaded to a file.



The Report Template is a pre-defined “canned” report supplied with Authentication Manager. The parameters that are needed to construct a query are not yet part of the template - the administrator supplies the desired parameters to transform the Template into a Query.

The template is the basis for customization into a query that will become named and saved in the system. Customization involves defining the required and optional parameters, the database to be queried, the sorting order of the output and the output table.

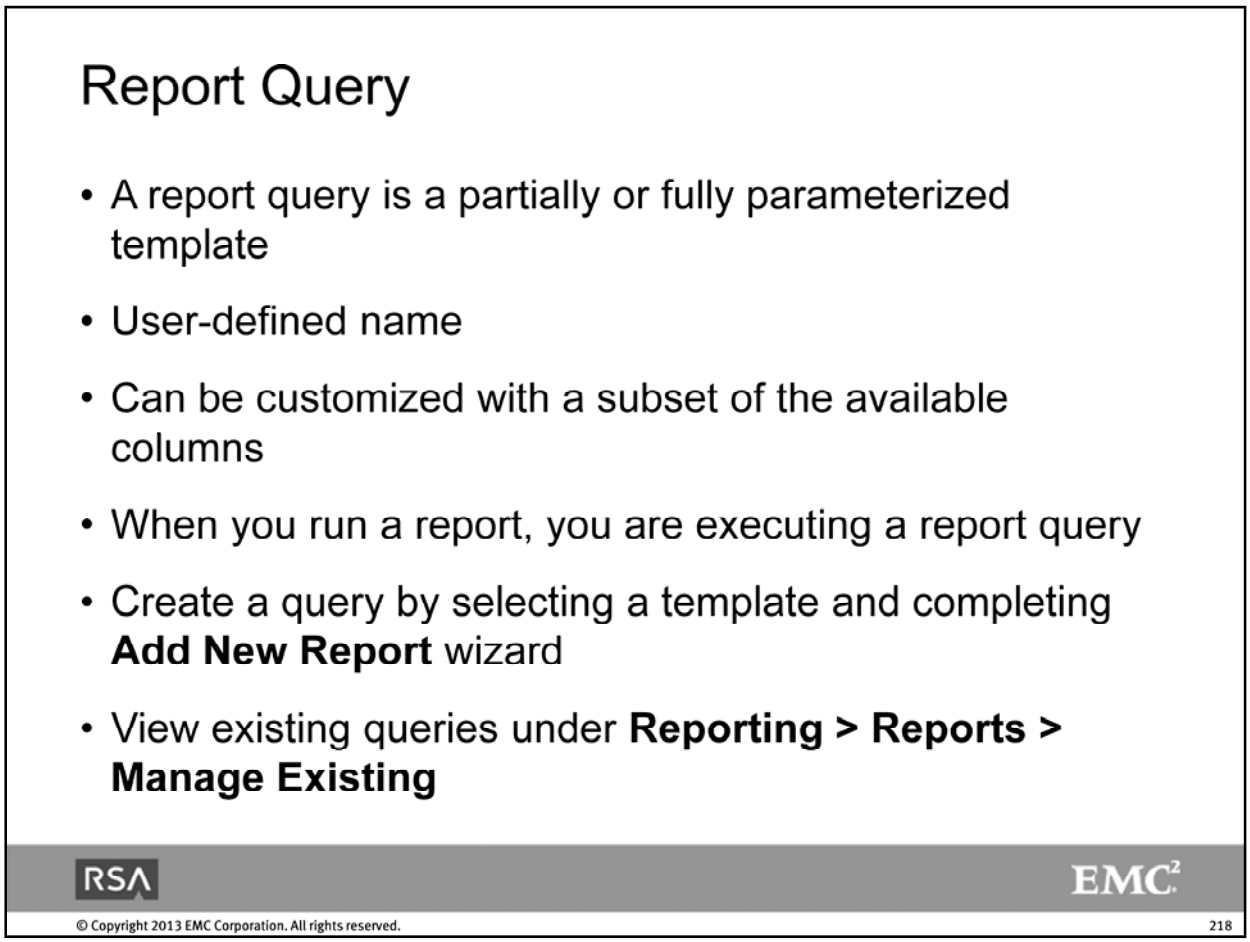
Some examples of available report templates are:

- Administrators with a Specified Role
- Administrator Activity
- Users with assigned RADIUS profile
- Users with Disabled Accounts
- Expired User Accounts
- Token Expiration Report

Report Template Examples

- Administrators with a Specified Role
- Administrator Activity
- Users with Disabled Accounts
- Expired User Accounts
- Token Expiration Report
- Users with assigned RADIUS profile
- *…many more… 31 report templates "out-of-box"*

RSA EMC²



A report query is a template with some or all of its parameters defined. A user-defined name and selection of display columns is required (by default, a query utilizes all available display columns).

Create a query by selecting a template and completing the form fields (in a wizard-like interface).

When you run a report, you are executing the report query with any of its customizations.

A query can be run as the owner (the report creator) or as the administrator who actually runs the report. This defines the scope to use when the report is run. By default, the data generated by the report is limited by the scope and permissions of the administrator who runs the report. The alternative is to choose the scope and permissions to be that of the report owner. (impersonating the owner).

For example, if an administrator runs a report as the report owner, and if the report owner has a broader scope than the administrator who runs the report, the report will include data from the broader scope, rather than from the more narrow scope of the administrator.

A report query can only be modified by the owner of the report - this capability can not be assigned to another administrator.

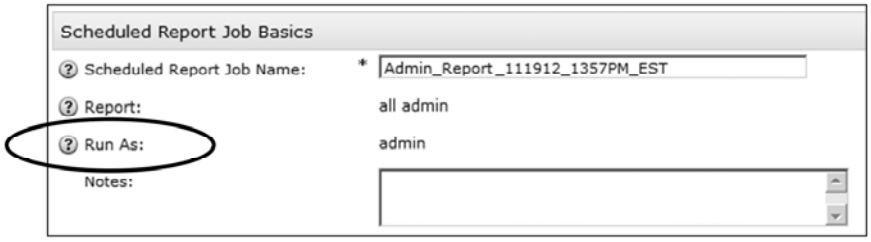
Queries are constructed with various parameter fields filtered using selection operators. These operators include:

- Starts with
- Ends with
- Contains
- Does not contain
- Is equal to

Others may use wildcard character (*) (selection lists, etc.).

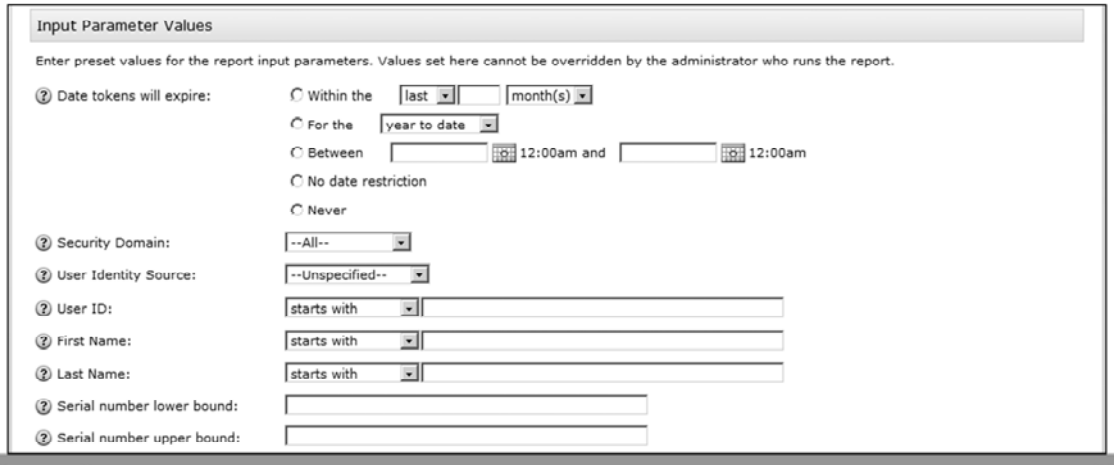
Report Query *(cont'd)*

- Query can be run as the owner or administrator
  - As owner by using the creator's access (impersonation)
  - As admin by using one's own access
- A report query can only be *modified* by the owner (name, selection parameters, etc)



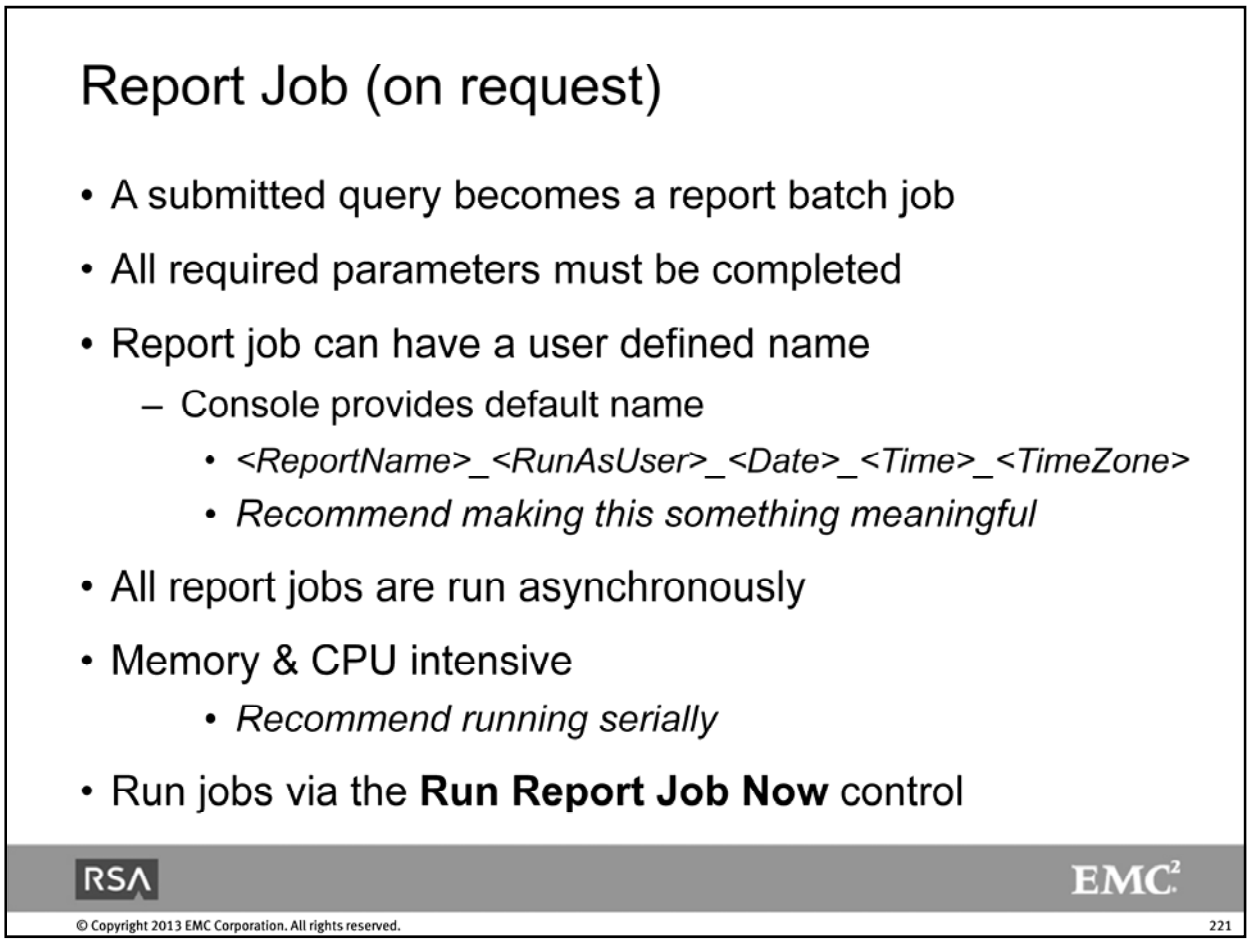
Report Query Parameters

- Selection operators include:
  - Starts with / Ends with
  - Contains / Does not contain
  - Is equal to
- Others may use wildcard character (*), selection lists, etc.



RSA EMC²

220



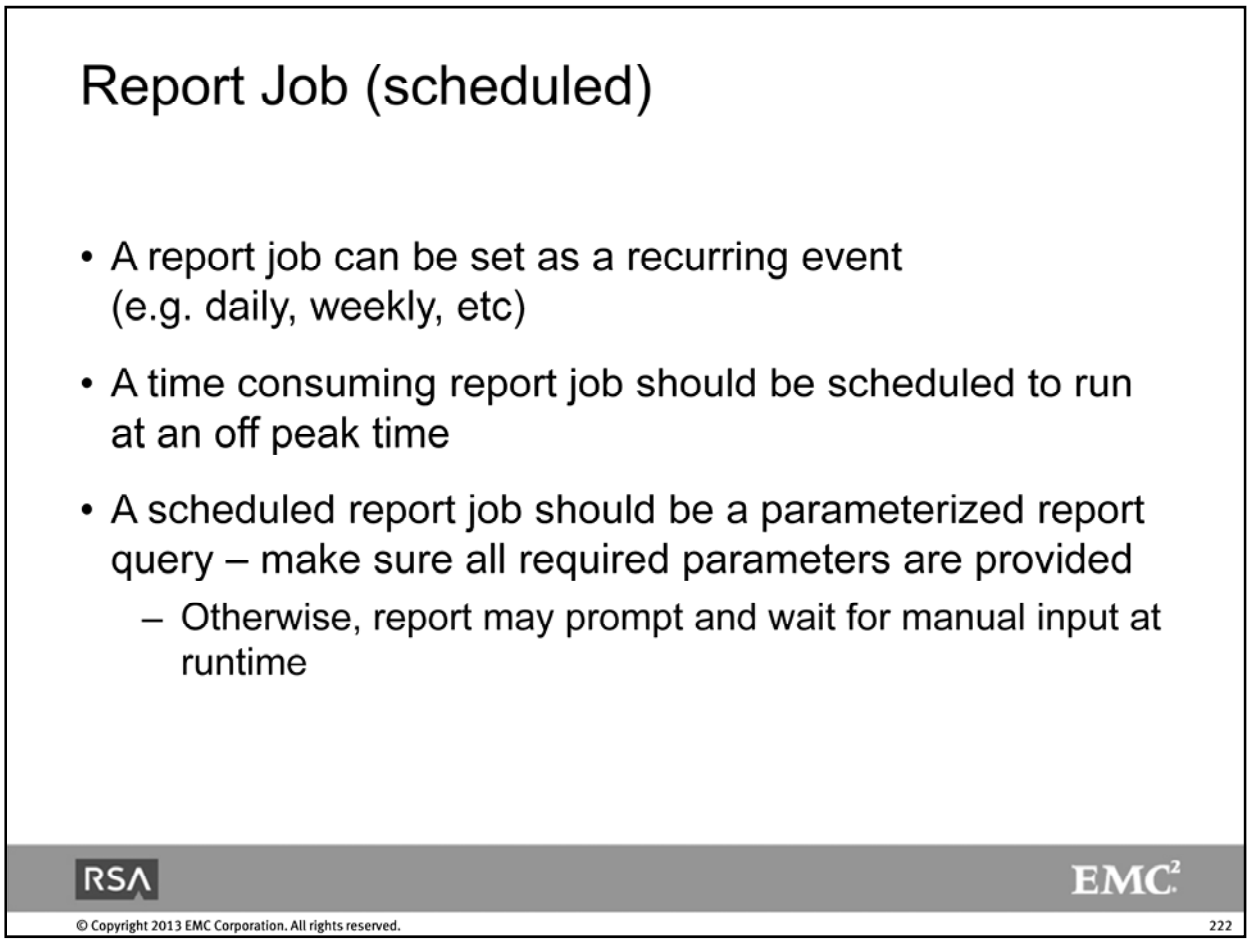
After a report query is constructed and saved, it can become a report job.

Report jobs can be run On Request or as Scheduled jobs.

An 'On Request' (or "Run Now') job should be ready to run with all required parameters specified. Additional filtering can be specified at the time the report is run.

An On Request report can be given a friendly name (by default the report is run with the query name and date/time values appended).

Reports are run asynchronously and are memory and CPU intensive. To lessen the burden on equipment, it is generally preferable to run reports serially.



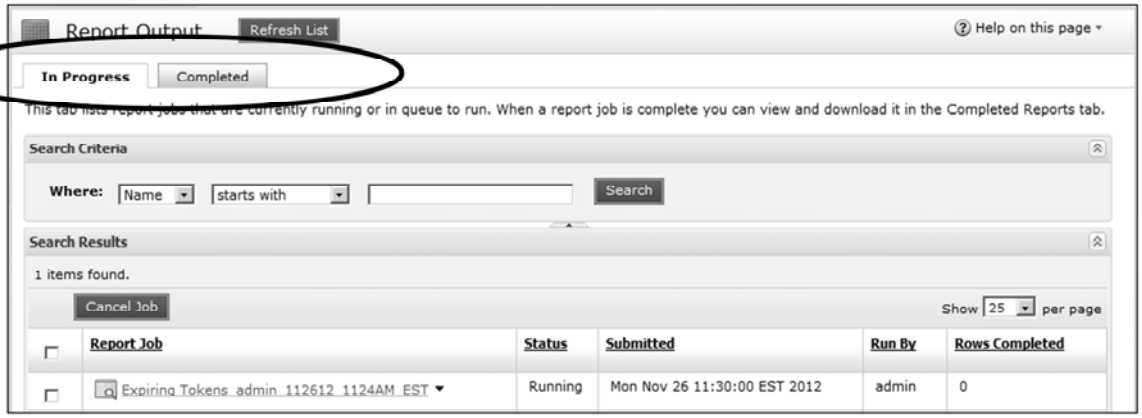
Scheduled jobs are set to run as recurring event (daily, weekly, monthly, etc.). These reports, too, should be ready to run with all required parameters specified. Additional filtering can be specified at the time the report is scheduled.

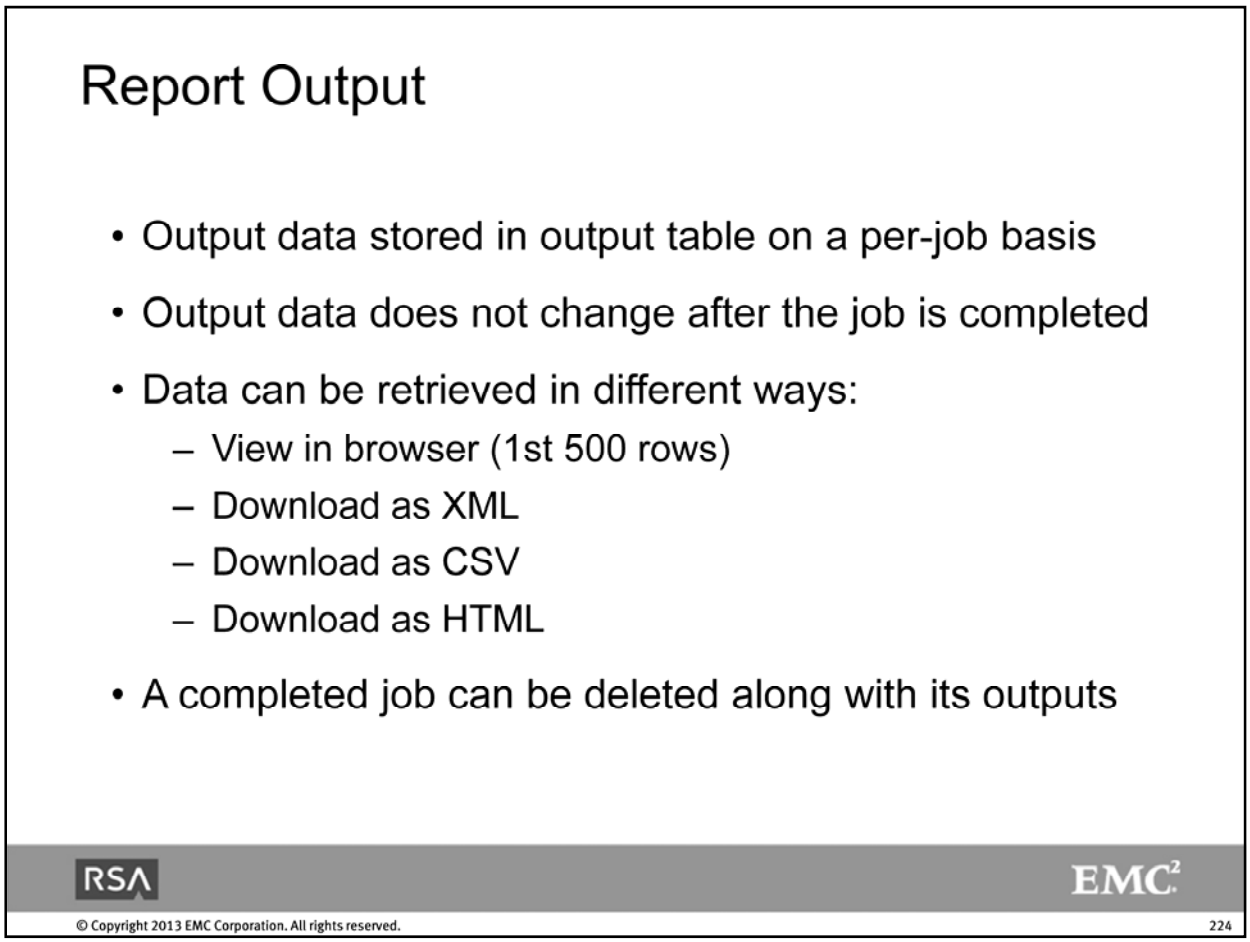
A report that is resource intensive and time consuming should be scheduled to run at an off-peak time.

The report job status screen (Report Output) shows reports that are Completed and In Progress and any additional details about the report.

Report Job Status

- Results may not be available instantaneously
- Console divided into In Progress and Completed tabs



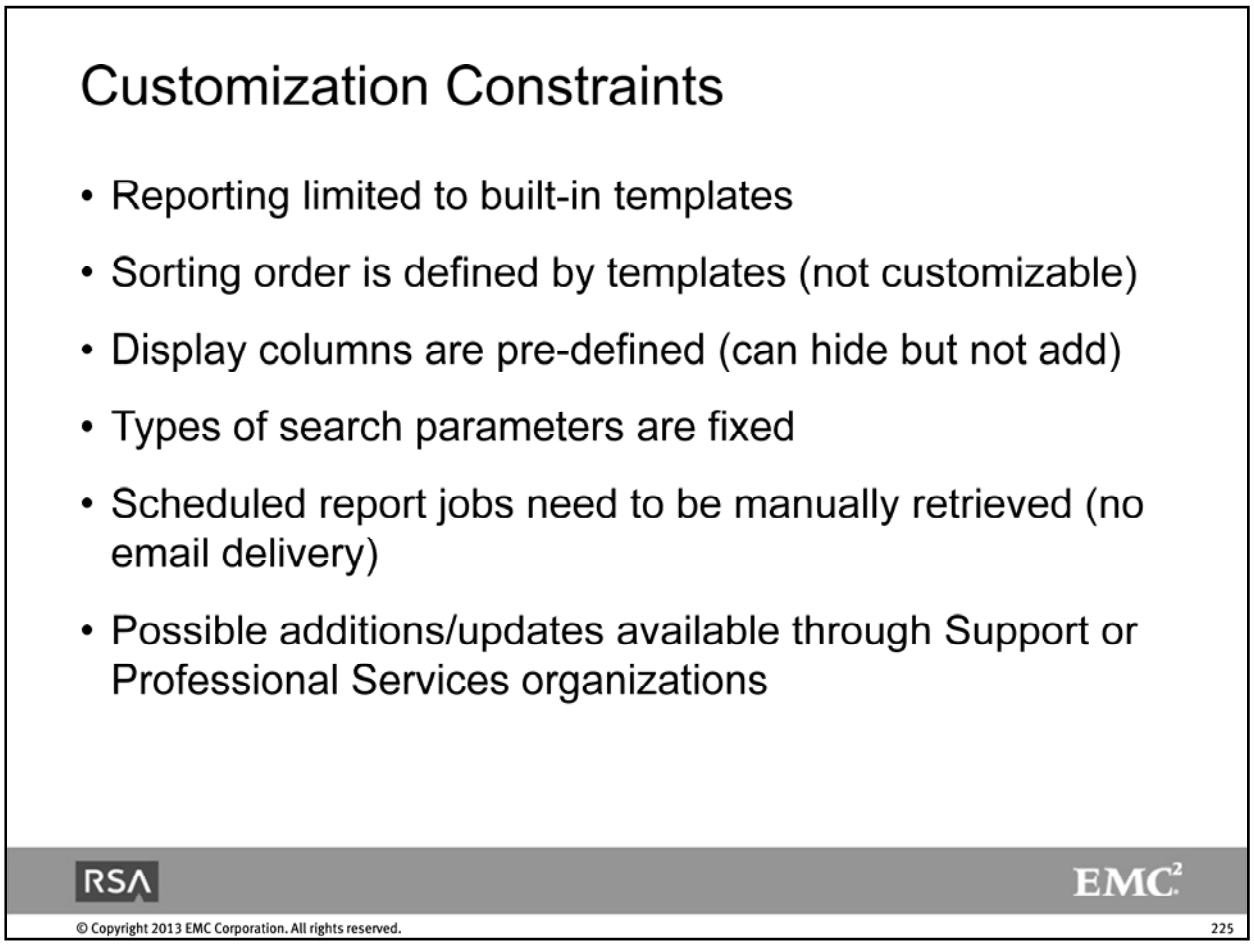


The output of a completed report is stored in the output table of the internal database on a pre-job basis. Once a report is run, its contents do not change.

Data from a completed report can be:

- Viewed in a browser (limited to the first 500 rows of data)
- Downloaded as an XML file
- Downloaded as a CSV file
- Downloaded as an HTML file

Once a job is complete, it may be deleted along with all of its outputs.



The reporting function does have some inherent constraints.

Reporting capability is limited to the templates provided in the software.

The sorting order in a report is defined by the template and cannot be changed by the report creator or person running the report (some flexibility is available in the report output through another tool such as Microsoft Excel).

Display columns are also pre-defined. you can choose which columns become part of the report from the list of available columns but you can not add more columns to a report.

The search parameters that you specify in a query are fixed - you can not add additional parameters.

When a report is run (as a scheduled job), the output requires manual retrieval - the report can not be emailed or set to be stored in another location.

As changes are made to the reporting functionality of Authentication Manager (additional templates, etc.), updates will be coordinated and made available through the RSA Support organization.

SQL Queries

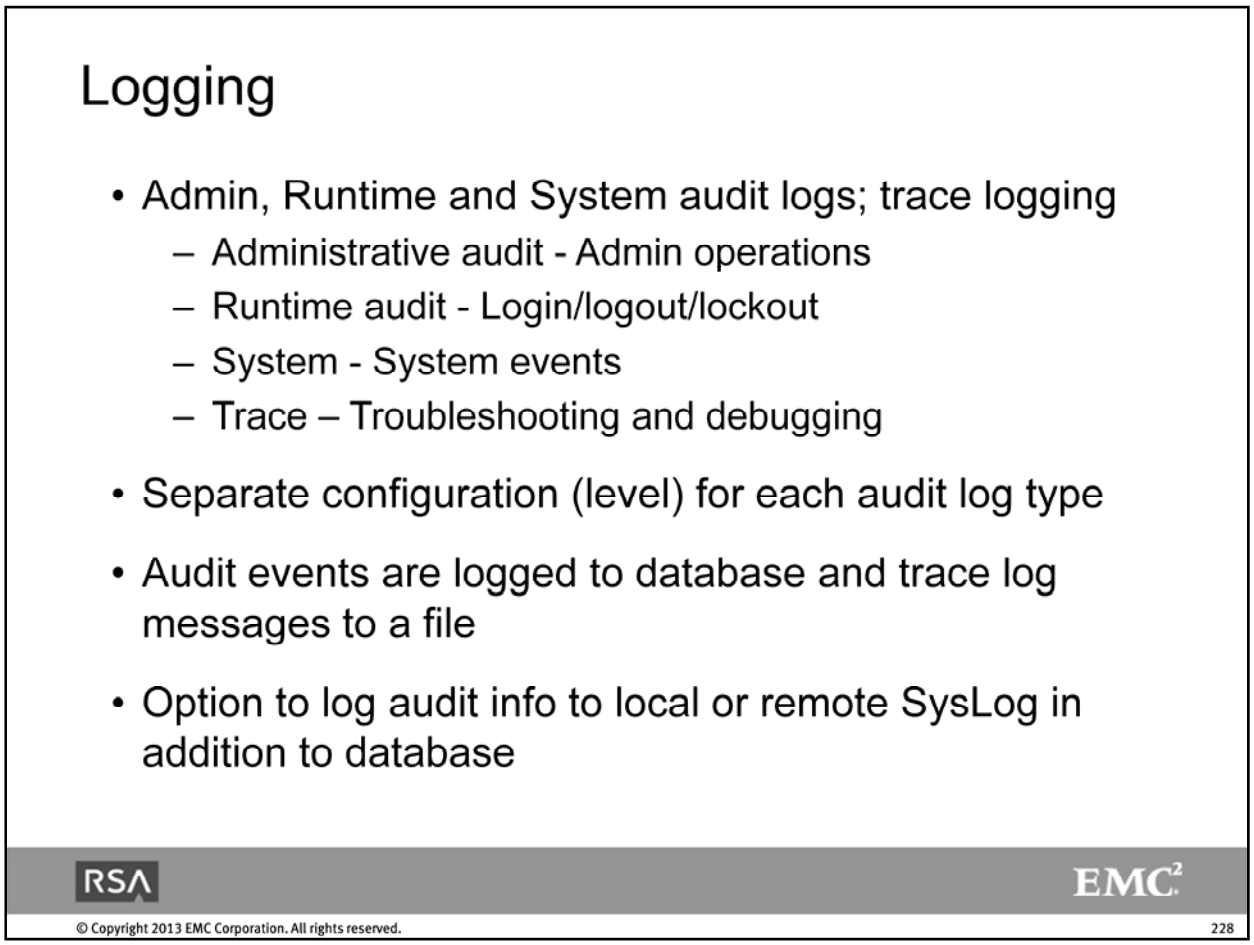
- SQL Queries to the database are supported by Authentication Manager developer tools
- Developer tools and documentation are available in the Authentication Manager distribution package
- Use involves an SQL driver for the database and using VBscript to execute queries
- Sample VBscript files are included in the SDK

You can create custom SQL queries from the RSA Authentication Manager internal database to supplement the report information available through the Security Console. Although you can create custom queries, this feature is not a replacement for the reports available in the Security Console.

Exercise: Reporting

- Create reports
  - On-Demand report
  - Scheduled report

RSA EMC²



Authentication Manager has three audit logging functions as well as a trace logging function.

The Administrative Audit Log logs all administrative operations in the system such as adding and editing users.

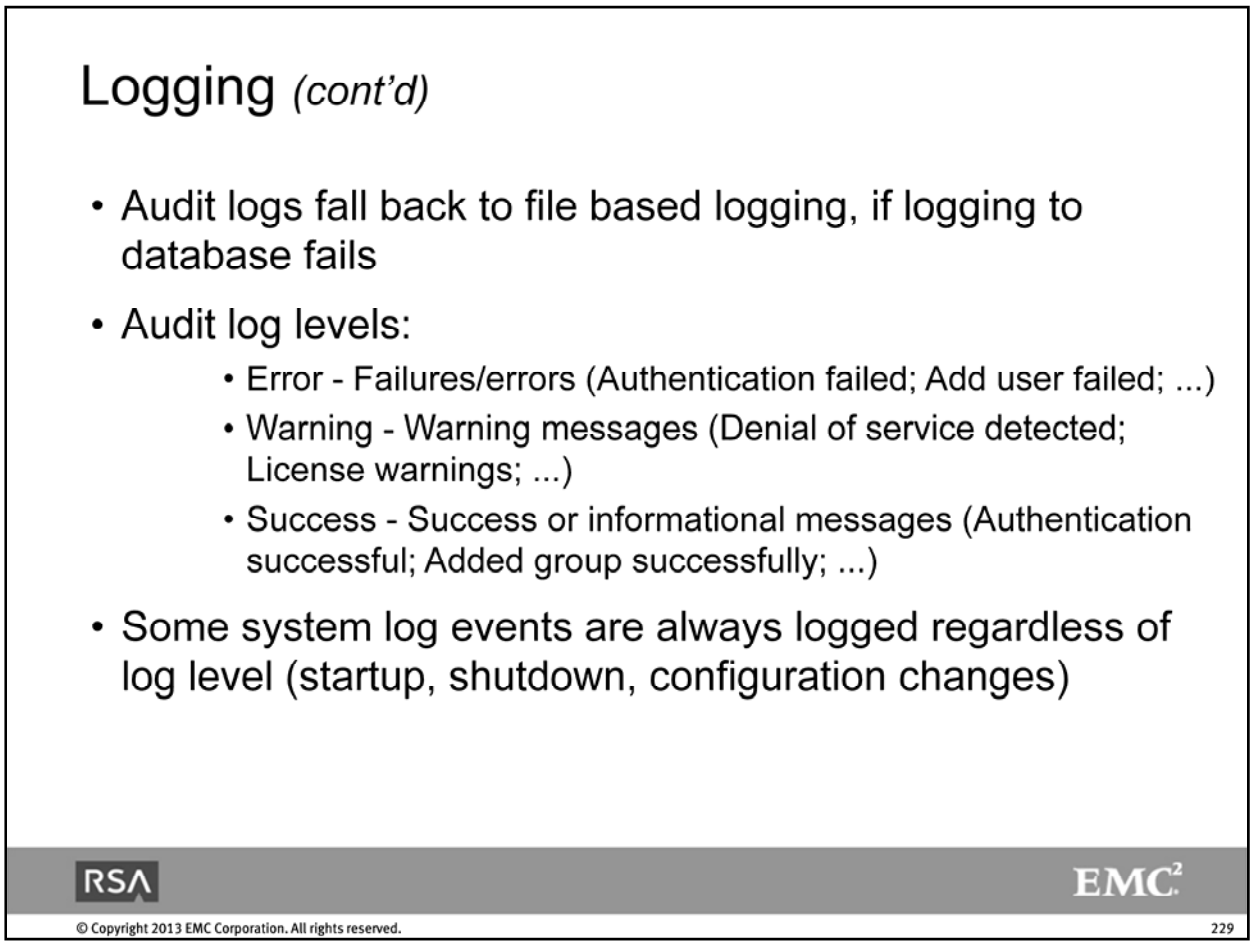
The Runtime Audit Log logs messages that record any runtime activity, such as authentication and authorization of users.

The System Audit Log captures log messages that record system level events such as "Authentication Manager Server started," and "Connection Manager lost db connection."

The Trace Log captures messages useful in debugging system issues.

Audit log events are logged to the Authentication Manager database while trace log events are written to a file.

In the event of a database problem, audit logs fall back to logging to a file so that information can continue to be captured.



Each of the logs' level of detail can be configured independently.

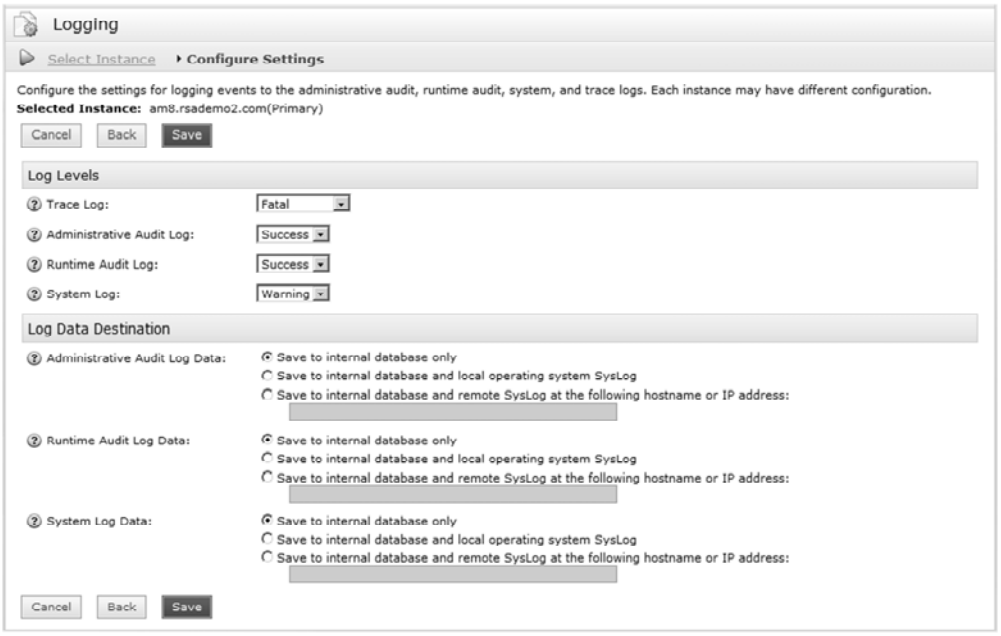
The following logging levels are available for Audit Logs:

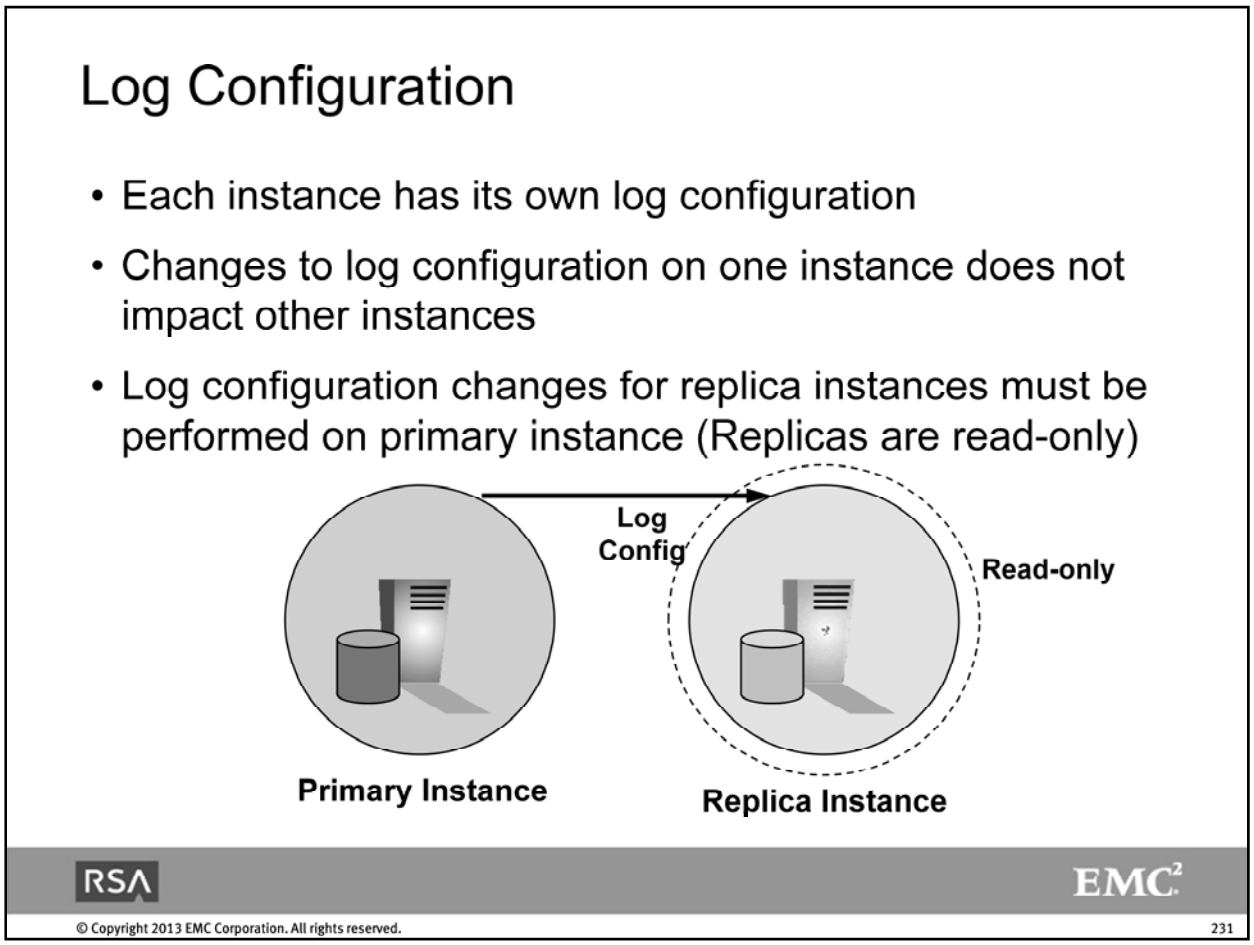
- None. No logging occurs.
- Error. Logs failures or errors in the system such as "add user failed".
- Warning. Logs warning messages such as "Denial of service detected" or license warnings.
- Success. Logs success or informational messages such as "Authentication successful" or "Added group successfully".

Some system log events are captured regardless of log level - Startup, Shutdown, Configuration changes.

Log Console

Admin console > Setup > Instances > Instance Name > Logging

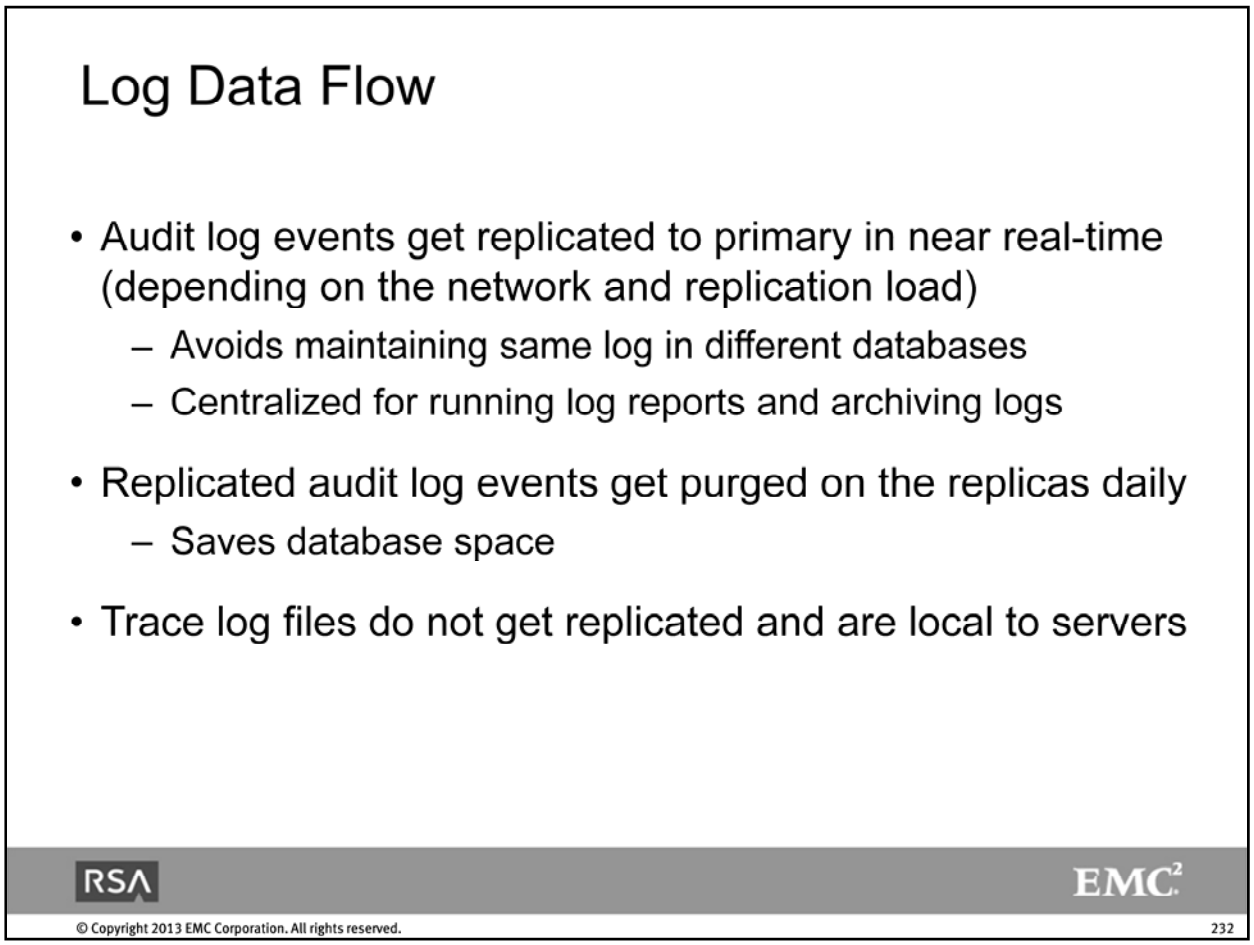




Each server instance has its own log configuration - changes to the configuration in one instance does not affect the configuration of another instance.

Each instance can have its own log configuration, which can be different for different instances.

Log configuration is performed in the Setup > Instances > Logging screen of the Security Console.

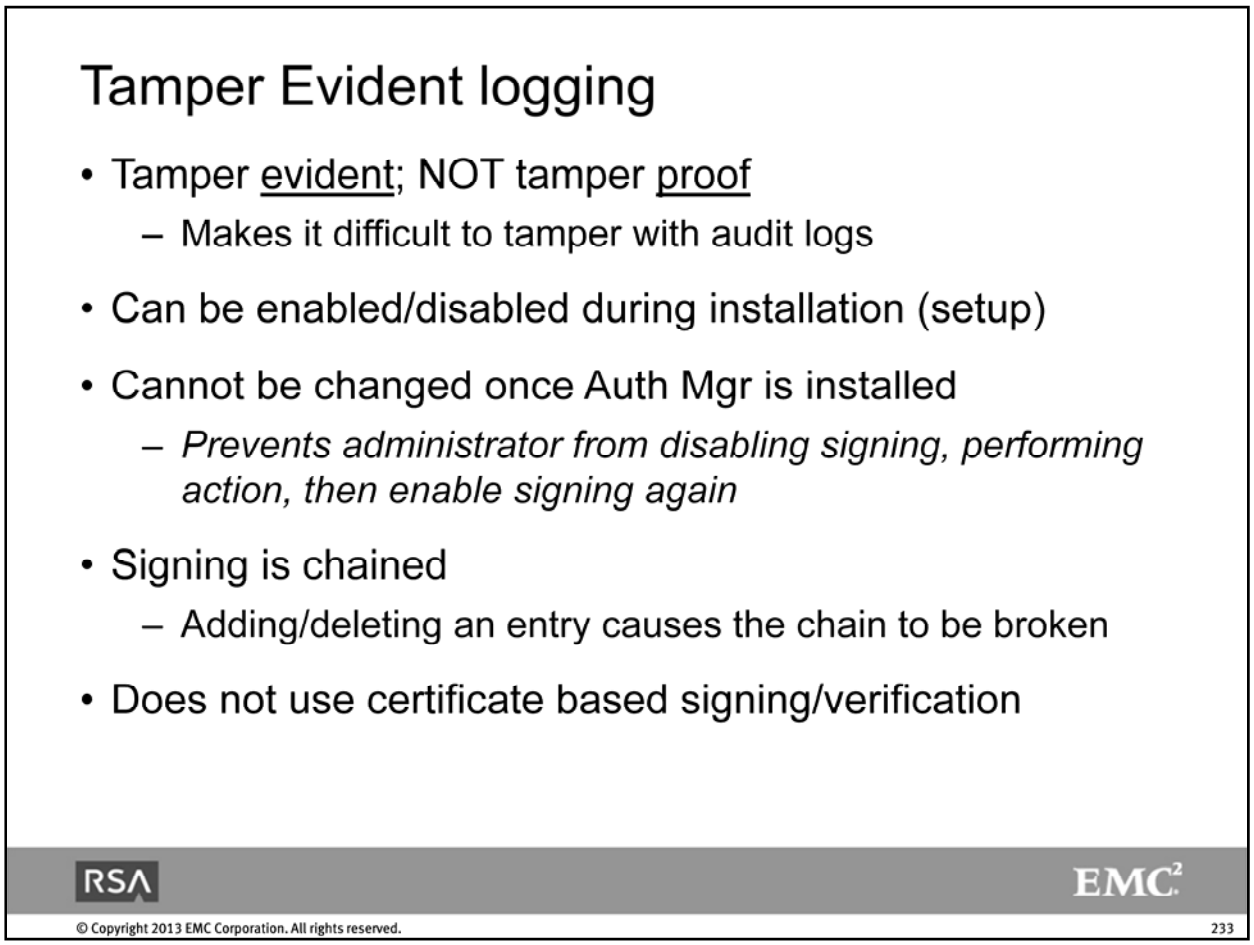


Audit log events are replicated to the Primary in near real-time (depending on the network and replication load). This avoids maintaining the same log information in different databases. By replicating information to the Primary, data is centralized for running log reports and archiving logs.

Replicated audit log events get purged on the Replica systems on a daily basis. This conserves database space.

Trace log files do not get replicated and are local to the server on which they are run.

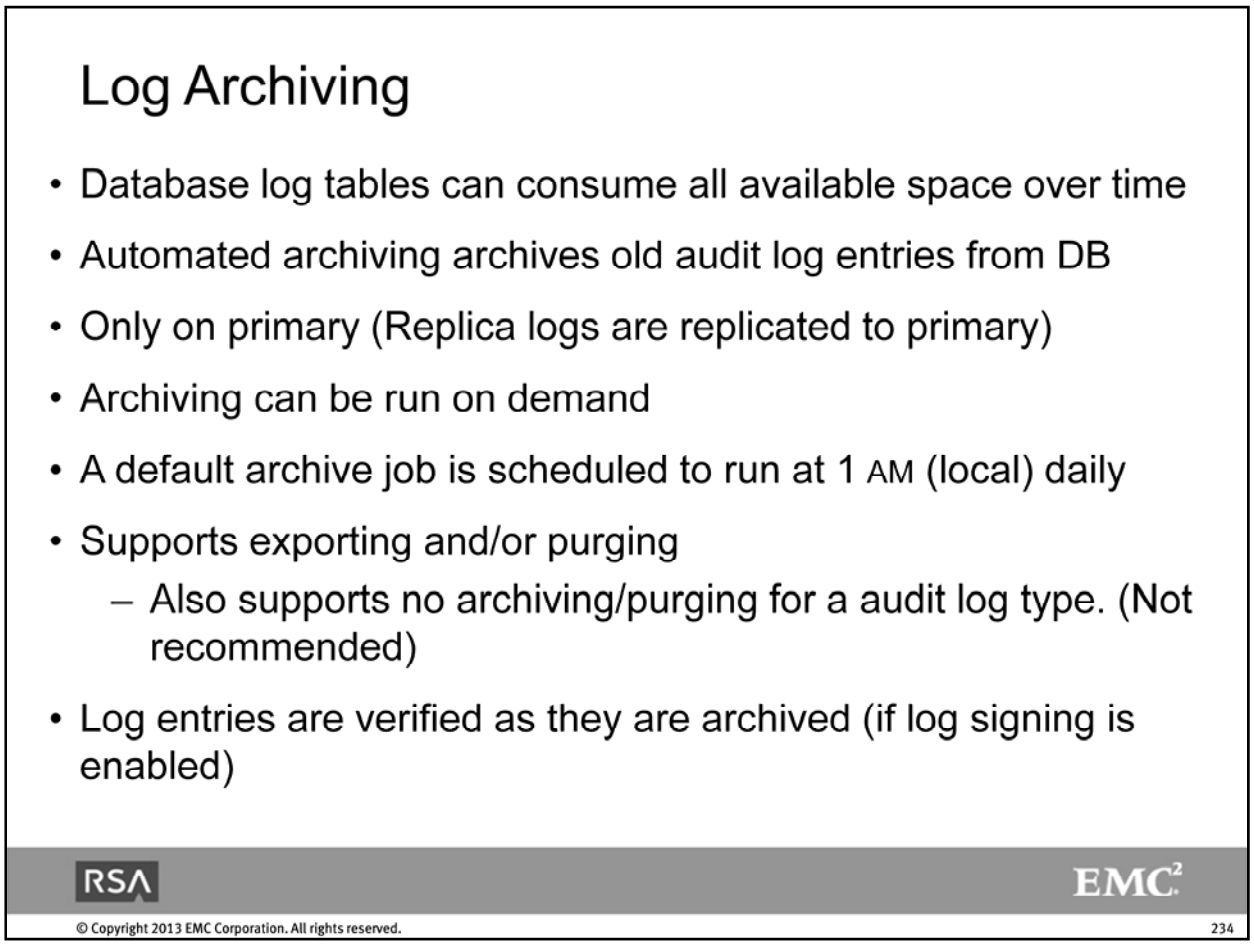
Log configuration changes for Replica instances must be done on the Primary instance (through the Security Console) as Replicas are read-only.



Authentication Manager logs are 'Tamper Evident' not tamper proof. This is not a rigorous, fool-proof system but it does make logs difficult to tamper with without detection.

The tamper evident function is enabled (or disabled) during installation and can not be changed once Authentication Manager is installed. This prevents the case where an administrator could disable log signing, perform some action, then re-enable signing.

The signing for logs is chained - adding or deleting a log entry caused the chain to be broken; thereby providing evidence of tampering. (The system does not use certificate-based signing or verification.)



Over time, log tables in the database can grow and consume all available space. Setting up automated archiving will archive old audit log entries from the database and restore available space.

Database archiving is only available on the Primary since Replica logs are replicated to Primary and Replica logs are purged daily.

Archiving can be run on demand or on a scheduled basis. A default archive job is scheduled to run at 2 am daily (server time).

Log Archiving *(cont'd)*

- If log signing is enabled, each archived log entry will have one of the following strings appended:
  - VERIFIED – Log entry intact and was not modified
  - TAMPERED – Log entry has been tampered with
  - UNKNOWN – Cannot determine if the entry is legitimate or tampered
- One archive log file is generated per audit log type per day
- Archived log files cannot be imported back into the database
- Each archived file is also signed
- Signature for each archived file is saved in a .sig file

RSA EMC²

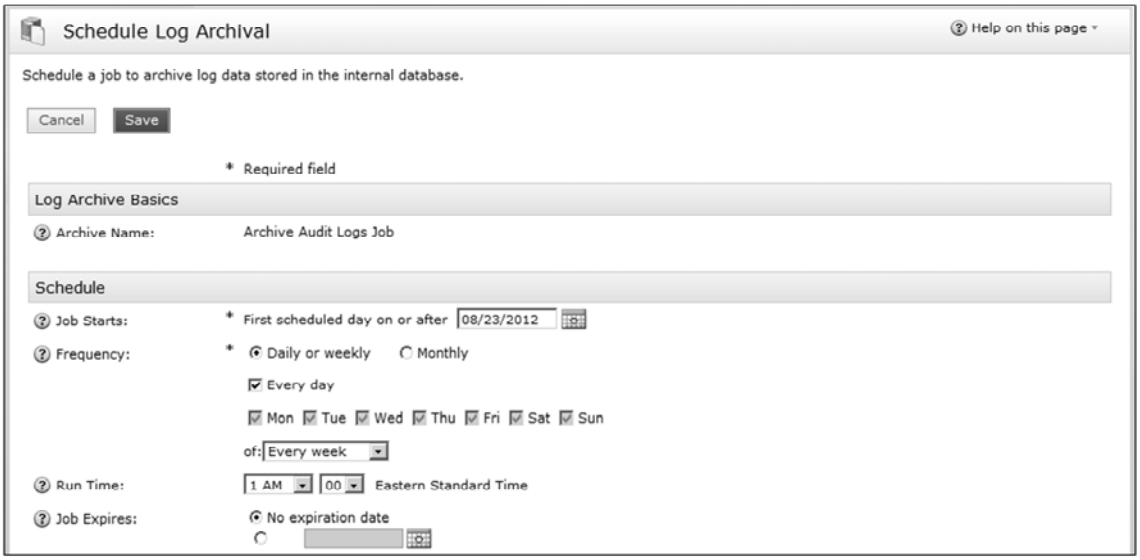
Log Archiving *(cont'd)*

- Archived log files are simple text files; entry fields are CSV format
- Can use text editor, Microsoft Excel or other reporting tools to view exported log files
- Archived log files are named after the audit log type and date:
  - audit_admin_< yyyy-MM-dd >.log
  - audit_runtime_<yyyy-MM-dd>.log
  - system_< yyyy-MM-dd >.log
- Log entries between 00:00:00 UTC and 23:59:59 UTC are logged to one file. Based on UTC time, not local host time

RSA EMC²

Log Console

Security Console > Administration > Log Management > Recurring Log Archive Jobs

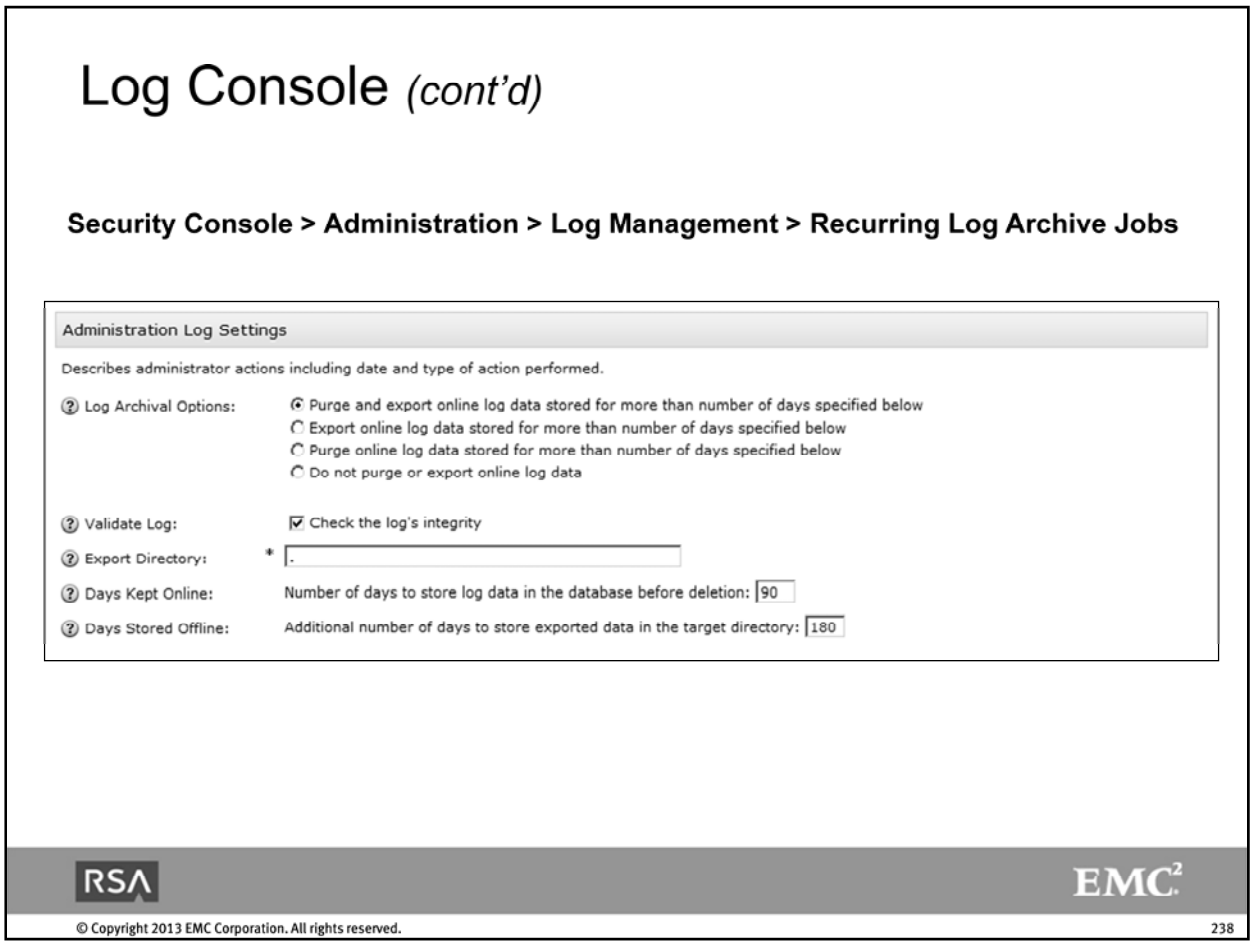


237

Recommended to run it every day at off peak hours

Can also be configured to run weekly/monthly or on specific days

Not recommended to use the job expiration date

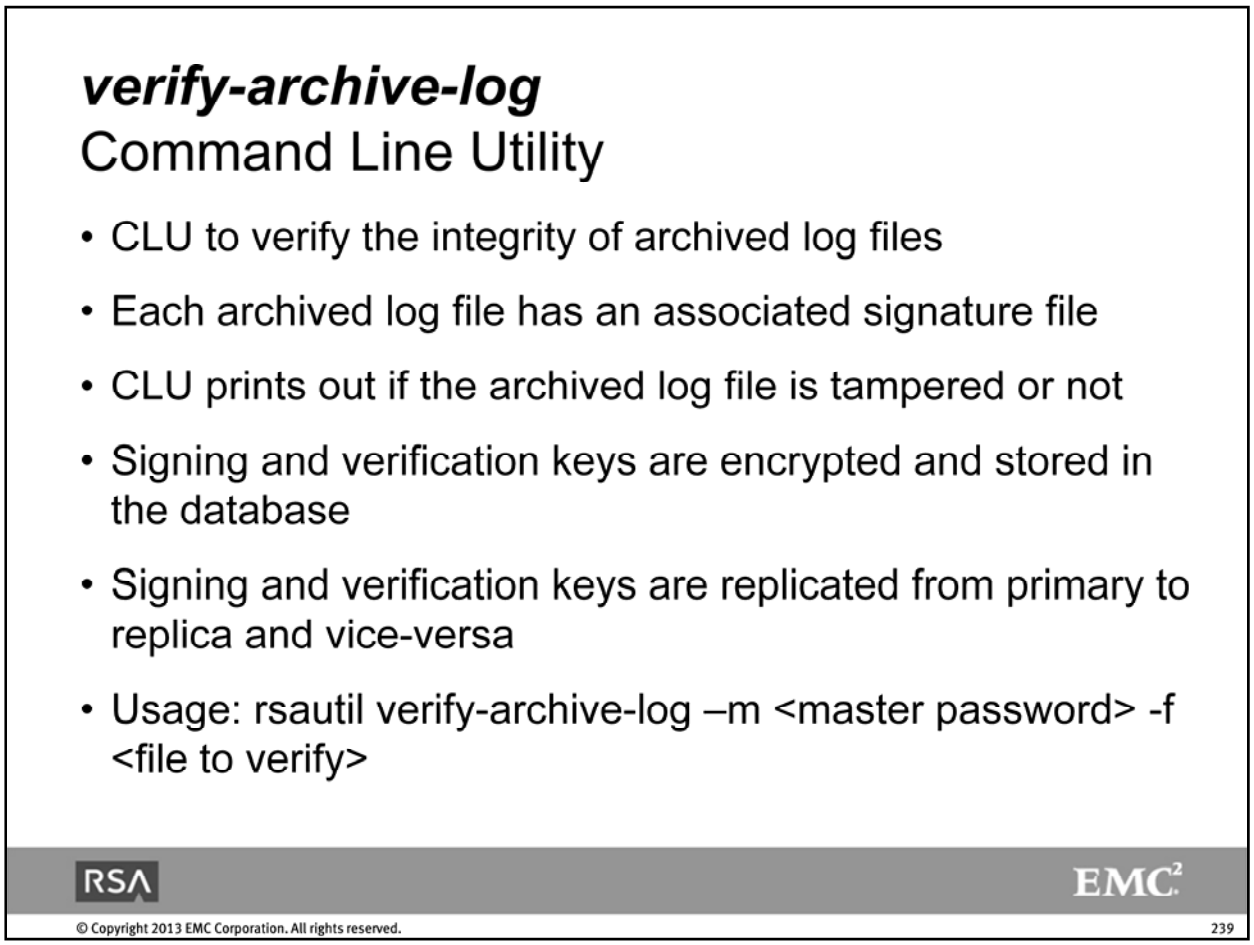


This is the setting for admin audit logs.

The same applies to runtime and system audit logs.

"Validate logs" field is only seen if log signing is enabled.

Export directory defaults to "." which is relative to <RSA_HOME>/server directory.



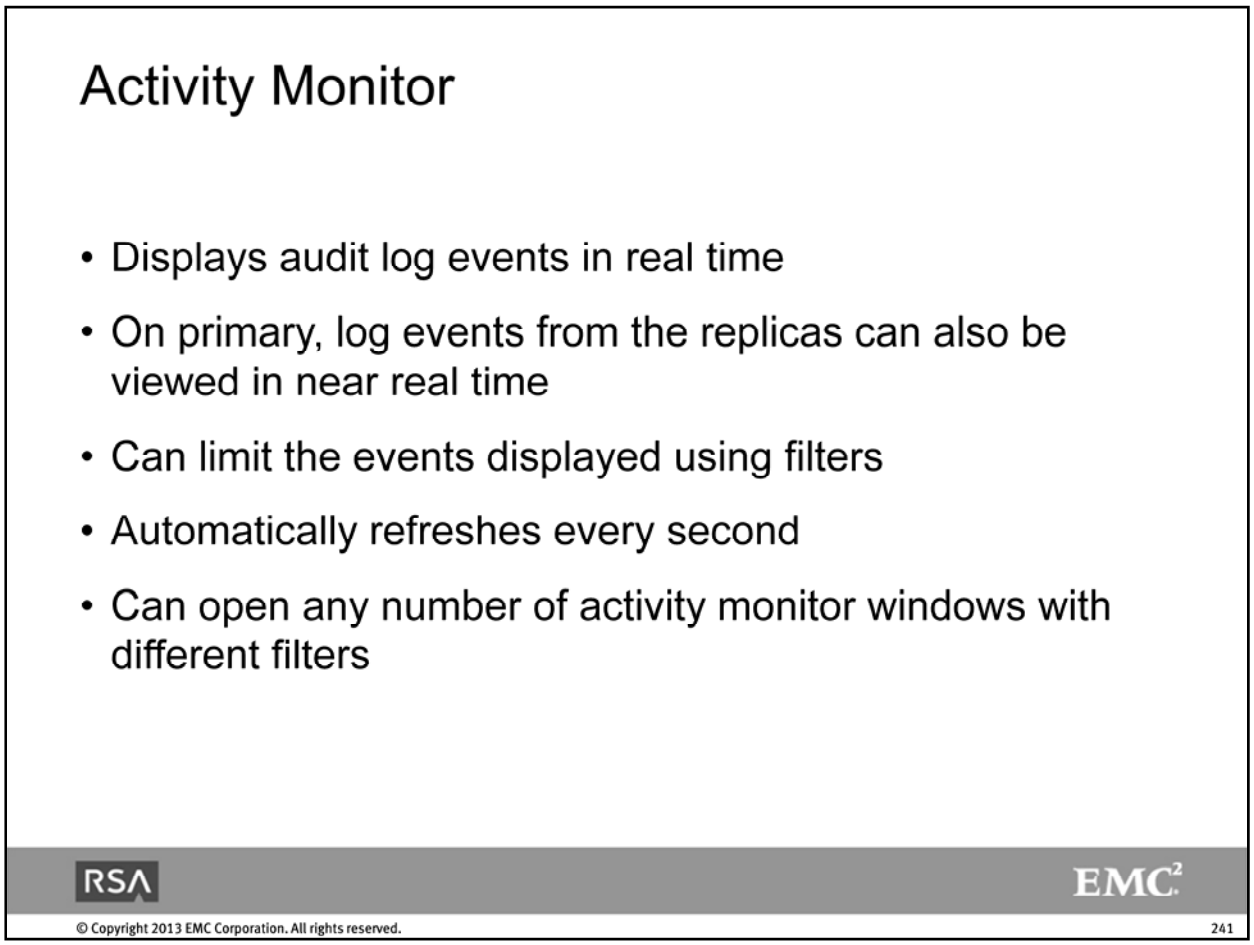
The verify-archive-log utility is a command-line utility used to verify the integrity of archived log files. Each archived log file has an associated signature file. The utility compares the archived file to the signature and prints out if the archived log file is tampered or not.

Signing and verification keys are encrypted and stored in the database and are replicated from Primary to Replica and vice-versa.

Exercise: Log Configuration

- Set Log Levels
- Set archiving job for logs

RSA EMC²



The Authentication Manager Activity Monitor displays audit log events in real time in a browser window. The monitor automatically refreshes every second.

On the Primary, log events from the Replicas can also be viewed in near real time (there is a small delay as Replica events are propagated to the Primary database).

You can limit the events displayed in the monitor using filters so that you can focus on a particular user or Agent in the system.

Any number of activity monitor windows can be opened at one time - each with different filters applied.

You can start and stop the event refresh in the monitor window at any tim - when paused, events are buffered and will display when the monitor is re-started. (The monitor will time out after an inactivity period 30 minutes - the same as the Security Console.)

A particular event can be selected to view more detailed information, if desired.

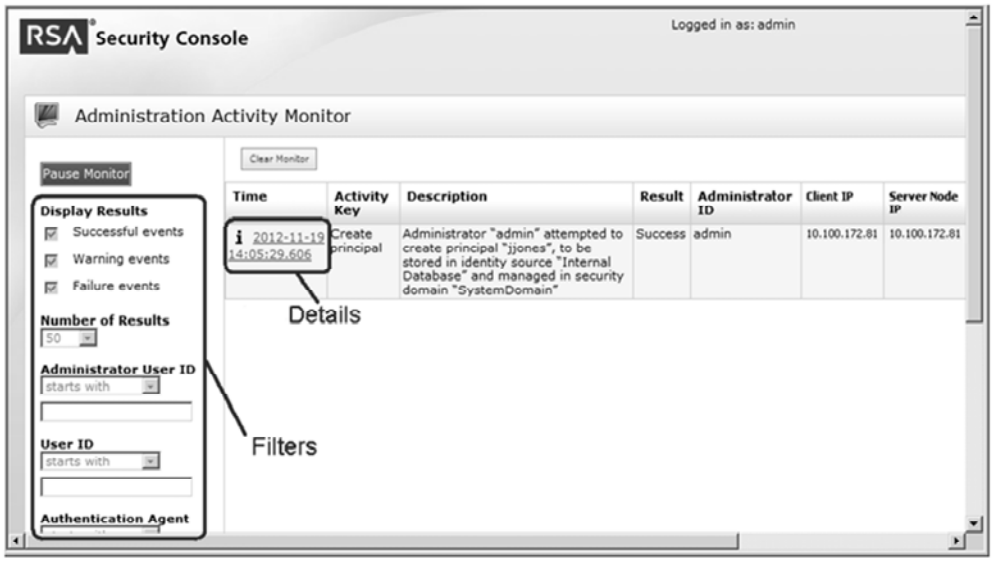
Activity Monitor *(cont'd)*

- Supports start/stop, pause/resume
- Buffers events when paused
- Inactivity timeout of 30 minutes (same as console)
- Individual events can be selected for additional detailed information
- Administrator can view events in their realm

RSA EMC²

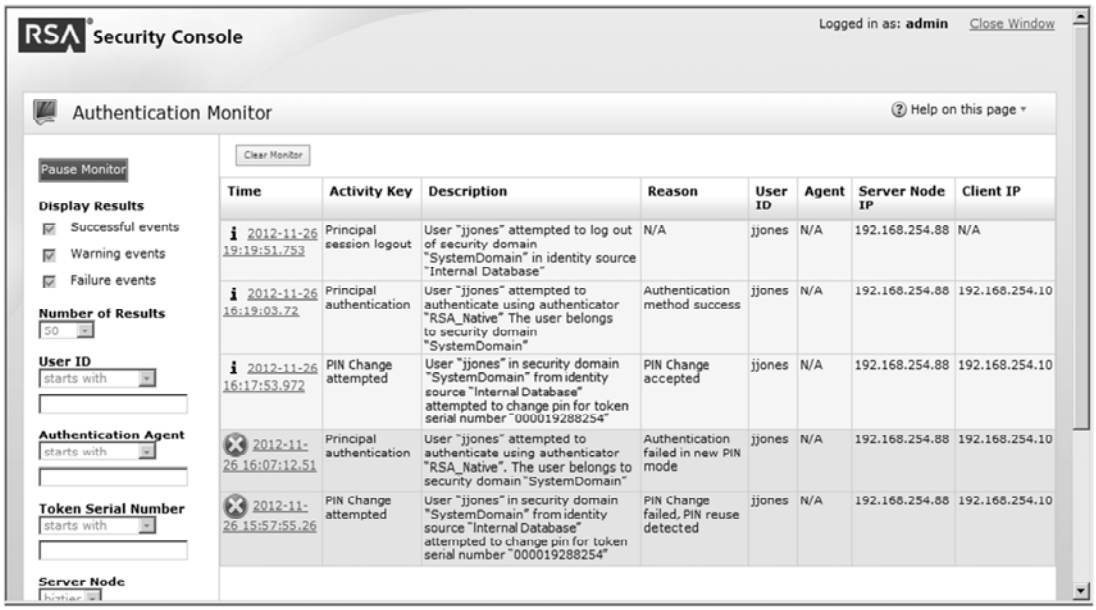
Activity Monitor *(cont'd)*

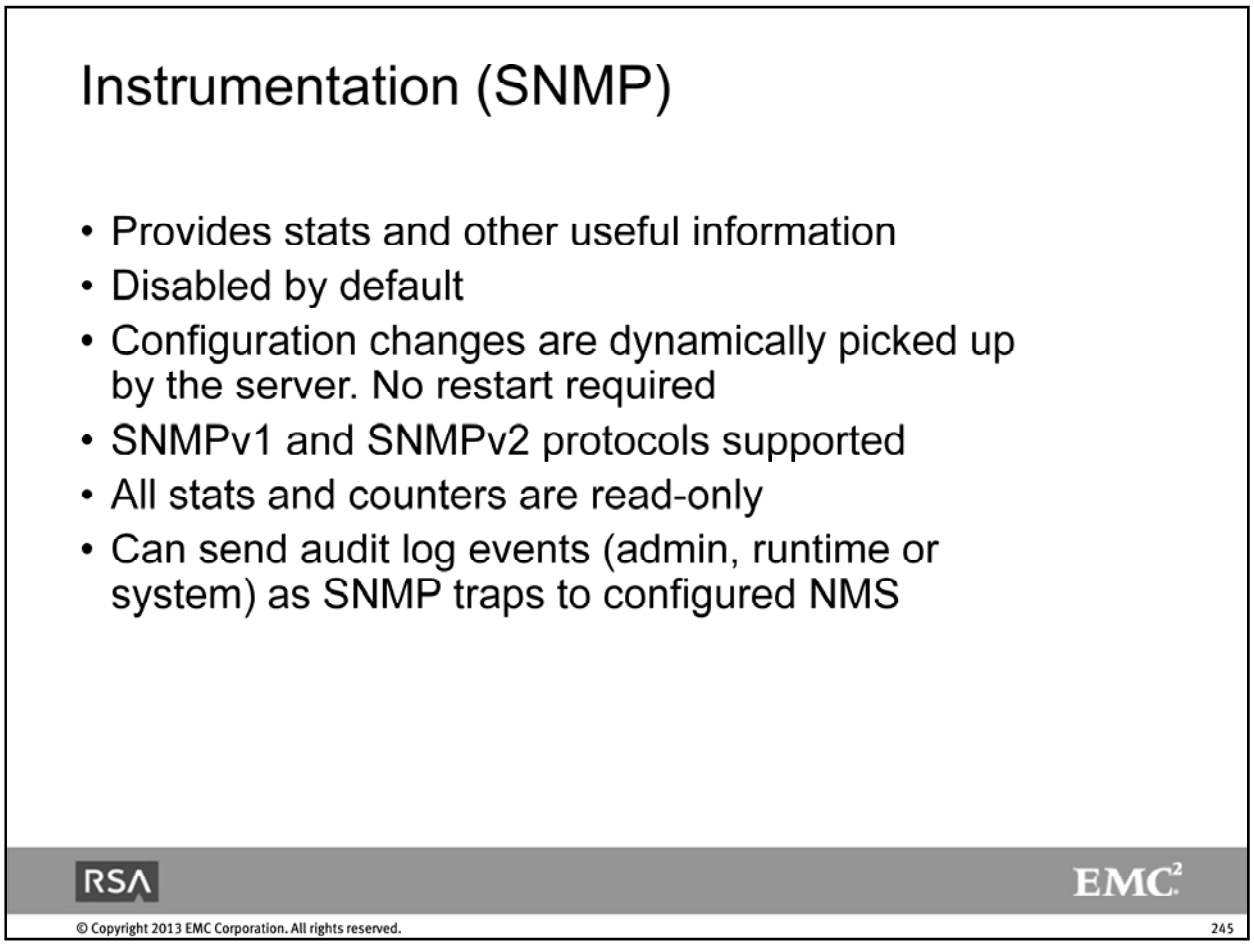
Security Console > Reporting > Real-time Activity Monitors > Administration Activity Monitor



Activity Monitor *(cont'd)*

Security Console > Reporting > Real-time Activity Monitors > Authentication Activity Monitor





Authentication Manager supports SNMPv1 and SNMPv2 protocols to provides statistics and other useful information to a Network Management System.

The SNMP function is disabled by default but can be configured through the Administration > Instance menu. All statistics and counters for the system are read-only.

Can send audit log events (admin, runtime or system) as SNMP traps to a configured NMS. SNMP trapping allows you to use the NMS to monitor error events occurring within Authentication Manager - when an error event occurs, Authentication Manager sends a notification to the NMS. Notifications can be intercepted and filtered based on the data sent in the trap message (message ID, for example).

Note: Authentication Manager sends event notifications for the fatal-level and error-level messages recorded to the system and audit logs.

Exercise: (Optional) SNMP Configuration

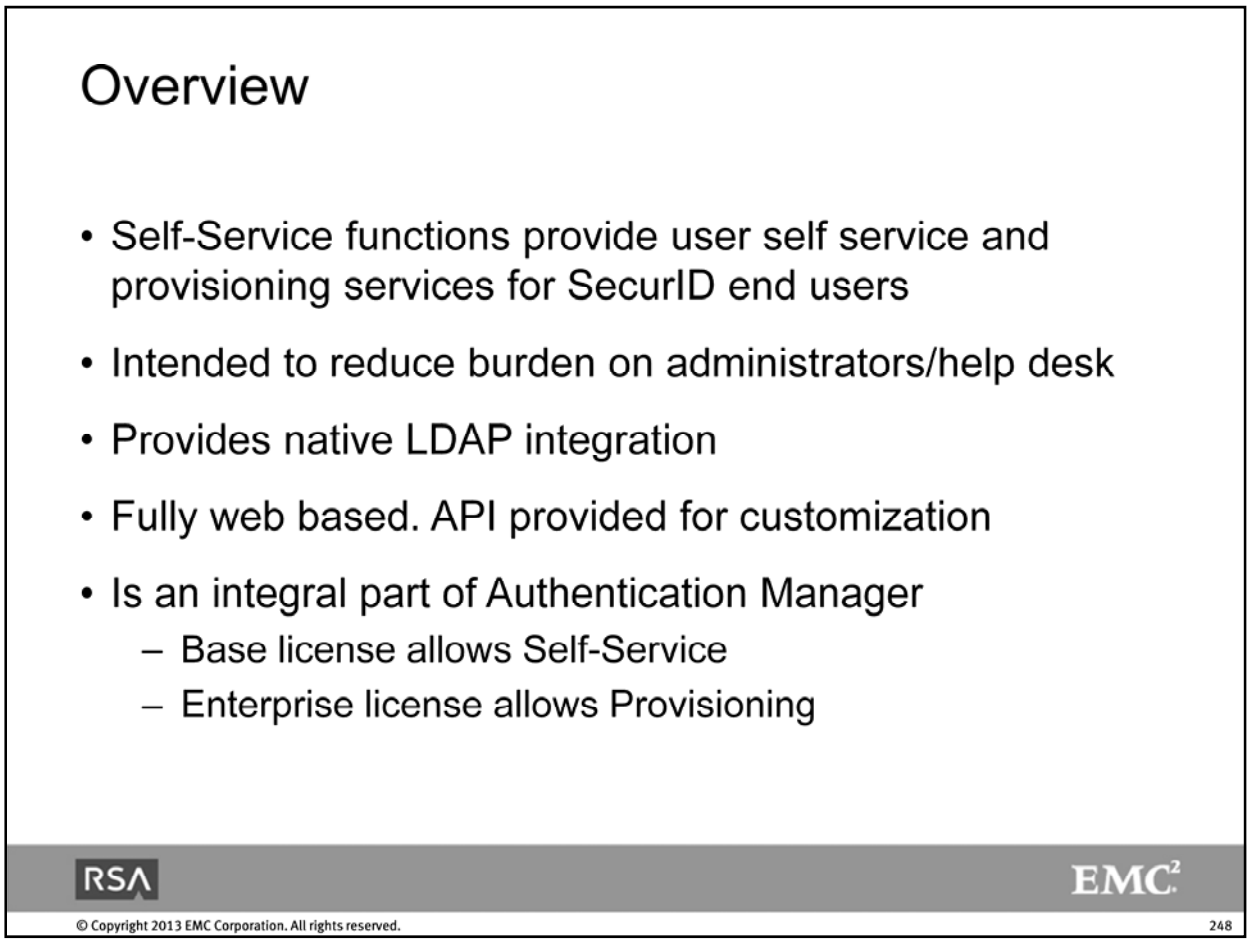
- View the configuration screen for SNMP

RSA EMC²

246

Self-Service

Unit 14

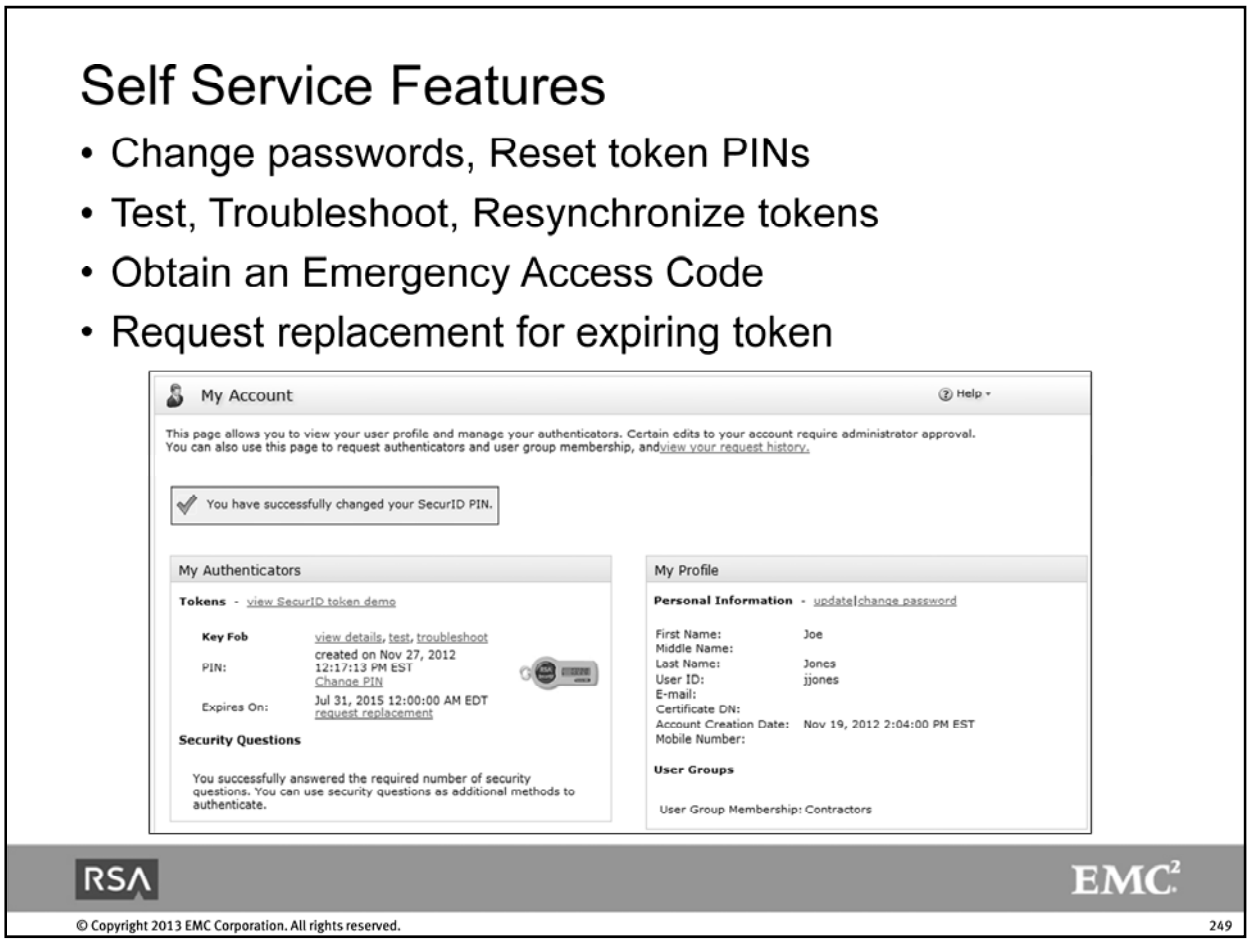


RSA Self-Service provisiioning is a web-based workflow system that provides user self-service options and automates the token deployment process.

Self-service allows you to reduce the time that the Help Desk spends servicing deployed tokens - when users forget their PINs, misplace their tokens, and require emergency access, or resynchronization. Users perform token maintenance tasks and troubleshoot tokens using the RSA Self-Service Console without involving administrators.

Provisioning streamlines the token deployment process if you are rolling out a large-scale token deployment. It also reduces administrative services and time typically associated with deploying tokens.

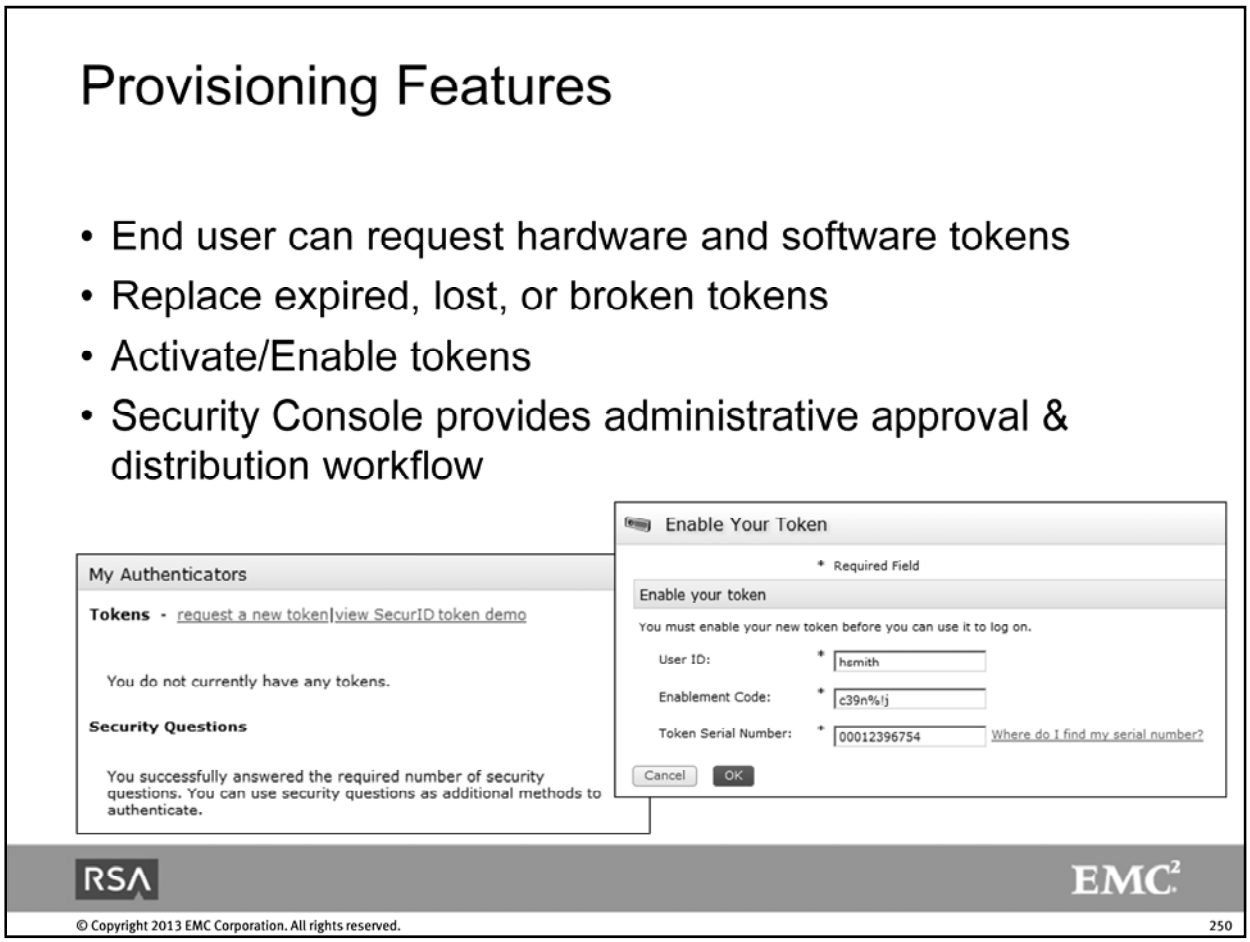
An Authentication Manager Base license allows self service; an Enterprise license allows provisioning.



The Self-Service Console is a browser-based interface where users can request tokens, troubleshoot tokens, and perform token maintenance tasks.

The following Token Management functions are available:

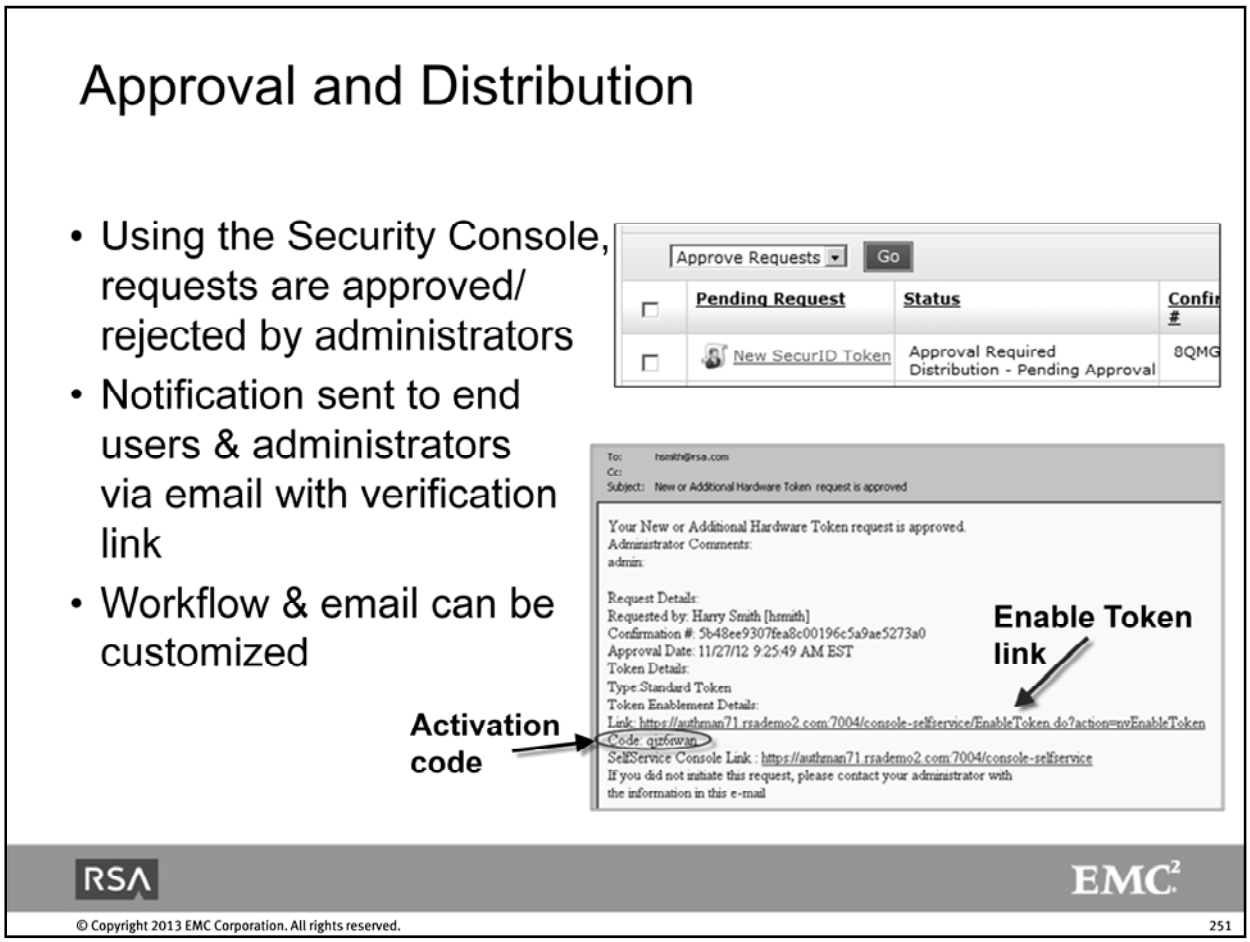
- Change passwords and PINs
- Resynchronize tokens
- Reset token PINs



With provisioning, users use the Self-Service Console to:

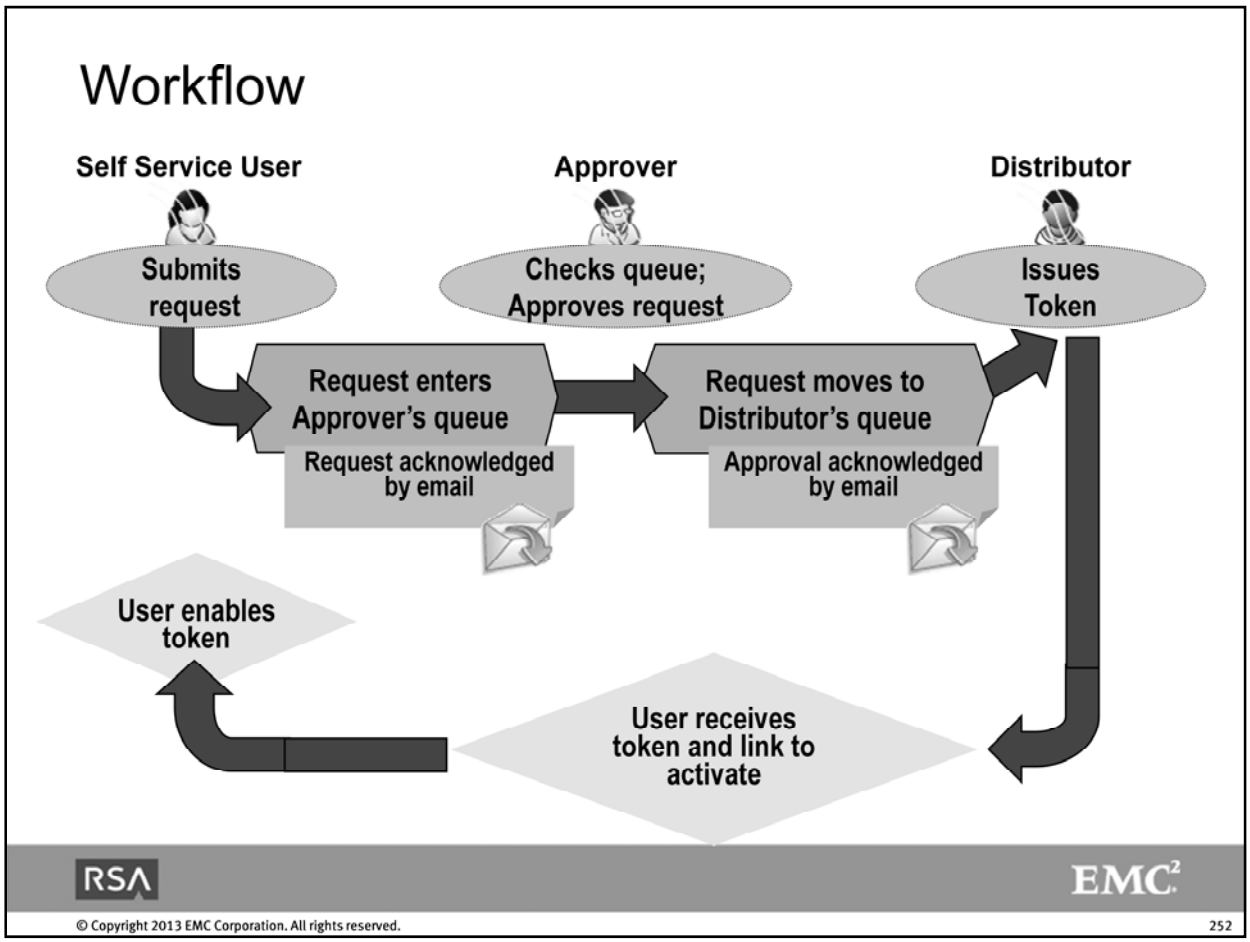
- Request hardware and software tokens
- Replace expired, lost, or broken tokens
- Activate tokens

The Security Console provides approval and distribution workflow management for the provisioning function.



Using the Security Console, requests are reviewed, approved or rejected by administrators.

Notification sent to end users and administrators via email. Both the workflow and email notifications can be customized.



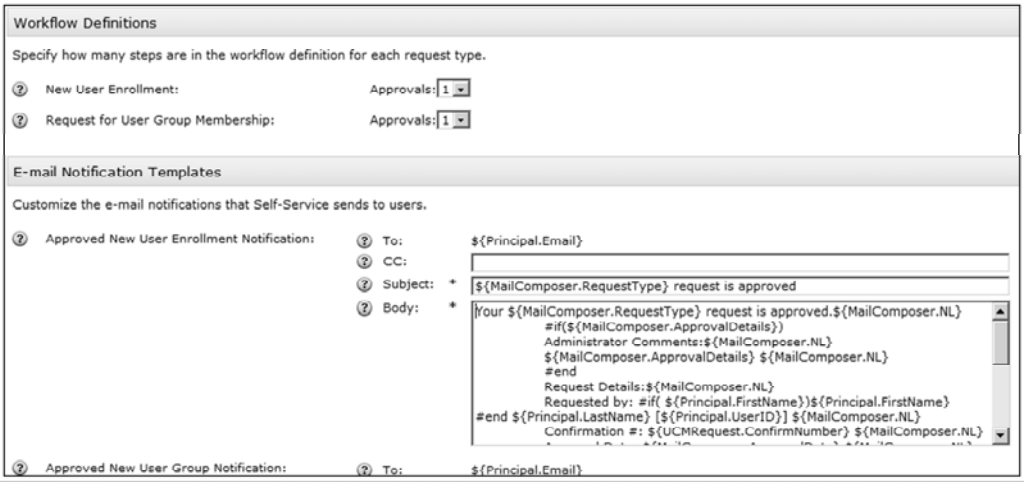
The workflow involves the following steps:

1. The user submits a request for a token
2. The request is entered into the Approver's work queue and the request is acknowledged to the user via email.
3. The Approver examines the request and approves or denies it. If approved (or denied) the result is acknowledged via email and the request moves to the Distributor's work queue.
4. The distributor issues the token to the user
5. Upon receiving the token, the user accesses the self-service console again to activate the token (using an activation code supplied in the approval email)

Workflow templates - configured by the administrator - determine the number of approval and distribution steps required in the process.

Workflow *(cont'd)*

- Workflow Definitions configured for self-service categories
  - User Enrollment, Token Request, On-Demand, etc.
  - Specifies number of approvals and/or distribution steps
- Email notification templates are accessed from same screen



Configuration

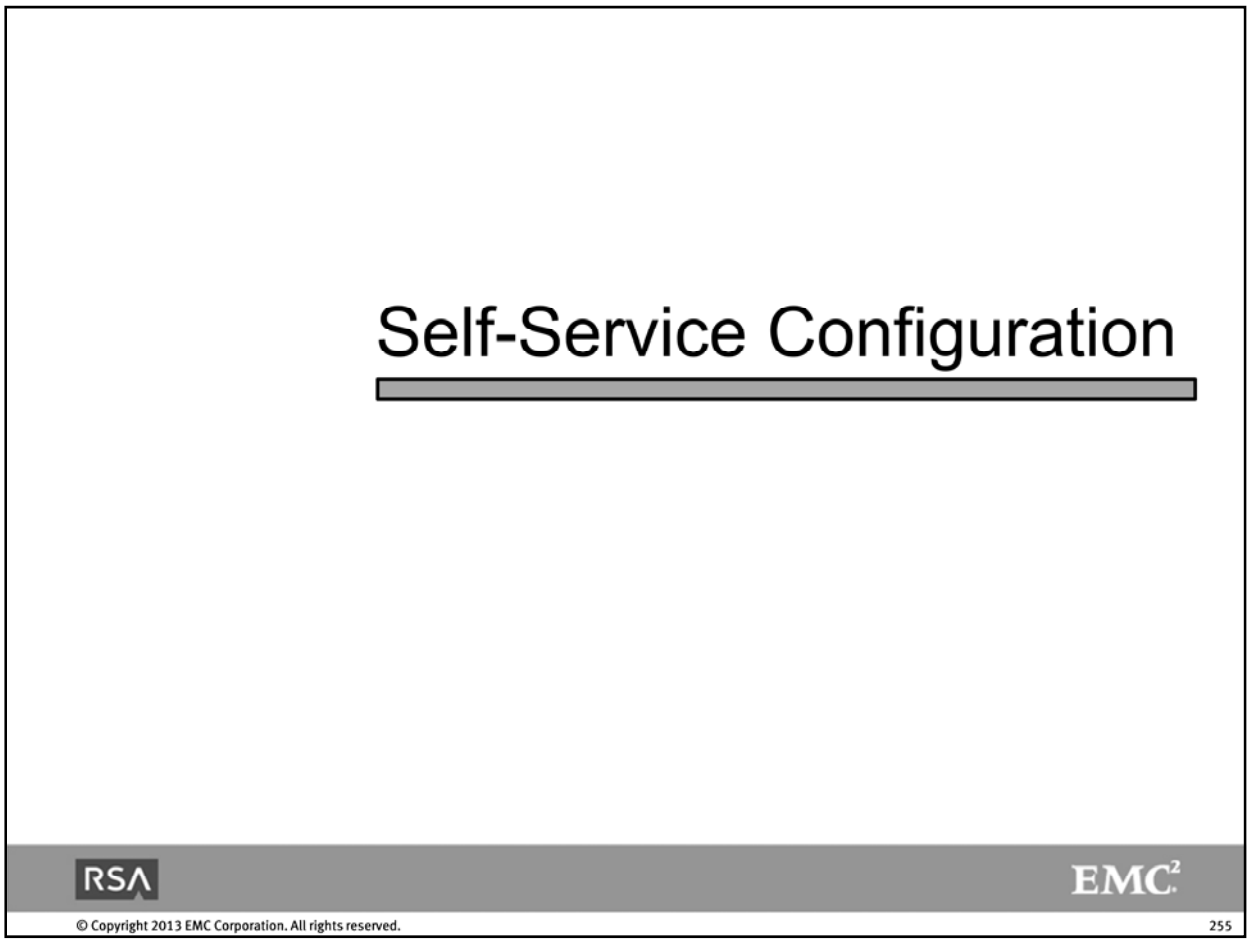
- Settings for Enrollment, Provisioning, and Customization
  - Select Identity Sources, Security Domains, and User Groups for Enrollment
  - Set workflow policies, authenticator, and emergency access parameters
  - Customize User Profiles, Self-service authentication, Email notifications, and Self-service features

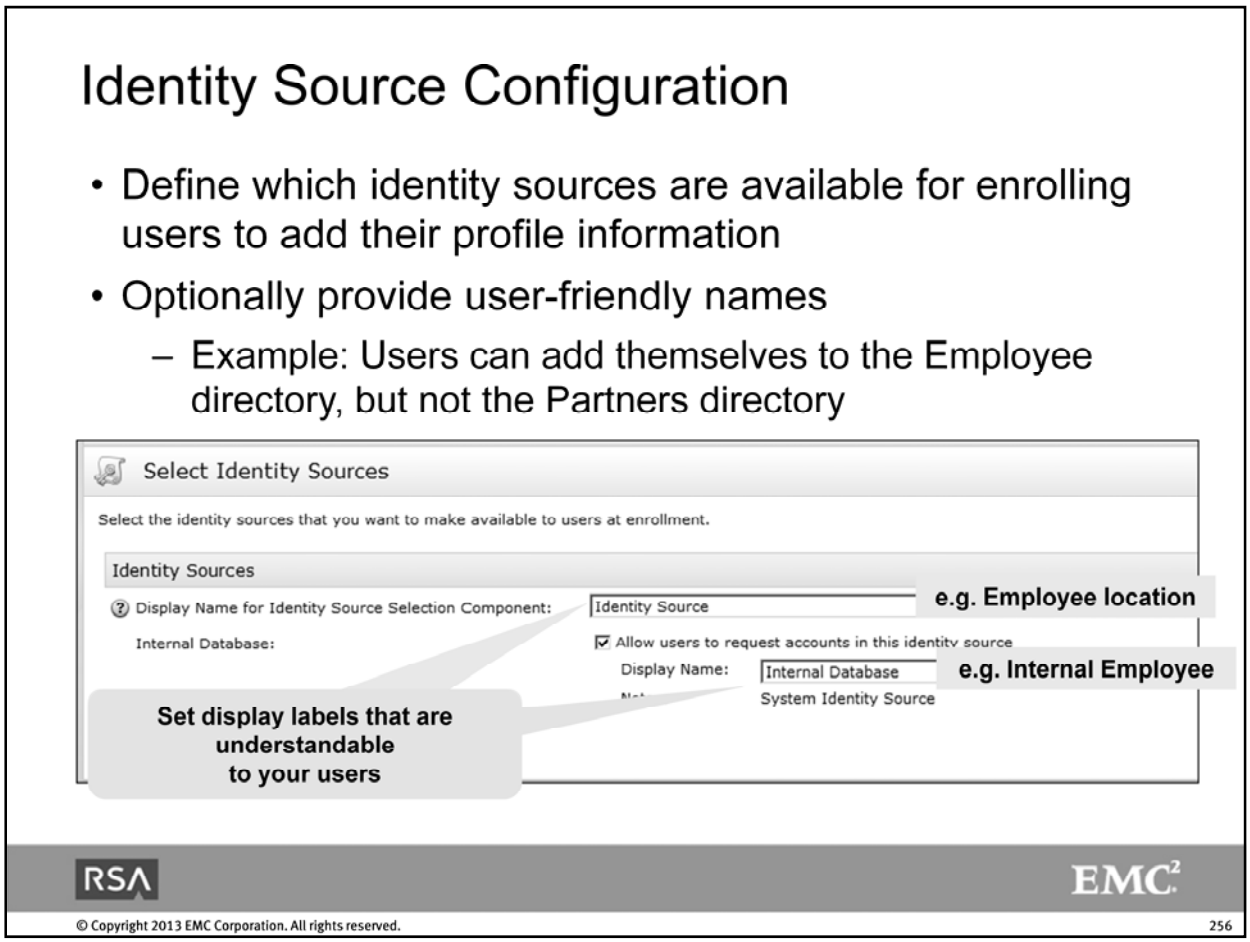
RSA EMC²

254

Configuration of Self-Service functionality involves the following:

- Defining the basic settings
- Defining the workflow
- Configuring email settings
- Selecting Identity Sources for enrollment
- Selecting Security Domains for enrollment
- Customizing user profiles
- Selecting User Groups for enrollment
- Managing token self-service

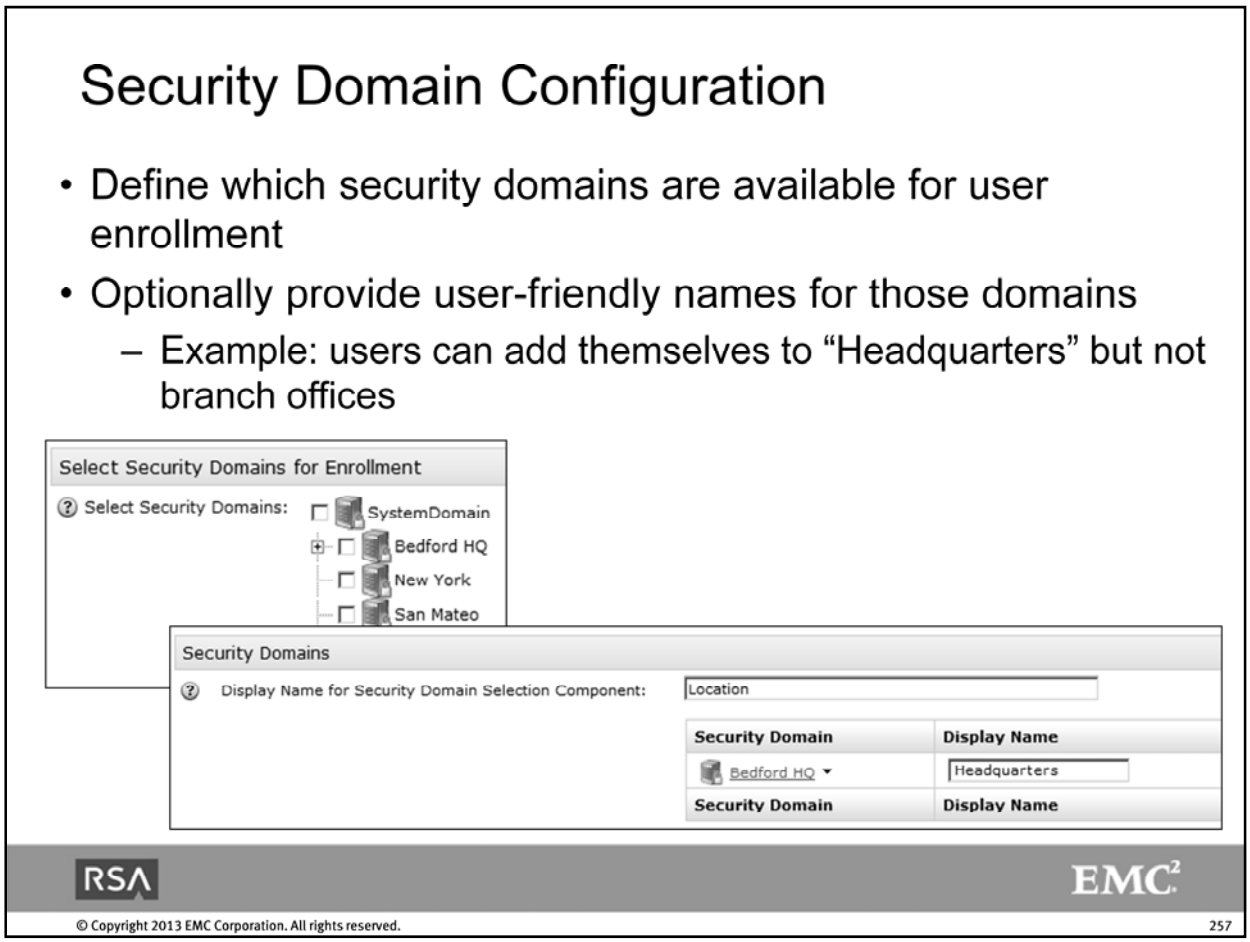

Self-Service Configuration
RSA
EMC²
© Copyright 2013 EMC Corporation. All rights reserved.
255



Identity Source configuration defines which identity sources are available for enrolling users to add their profile information.

User-friendly names can be configured for presentation to users to help them choose the correct identity source when enrolling.

Restrictions can be applied, for example, so that users can add themselves to the Employee directory, but not the Partners directory.



Security Domain configuration defines which security domains are available for enrolling users to add their accounts.

User-friendly names can also be configured for those domains.

The security domains that you make available for user enrollment determine which users administrators can manage, and limit the scope of the administrators’ control by limiting the security domains to which they have access.

Group Configuration

- Define which user groups are available for enrolling users to join
- Optionally provide user-friendly names for those selected groups

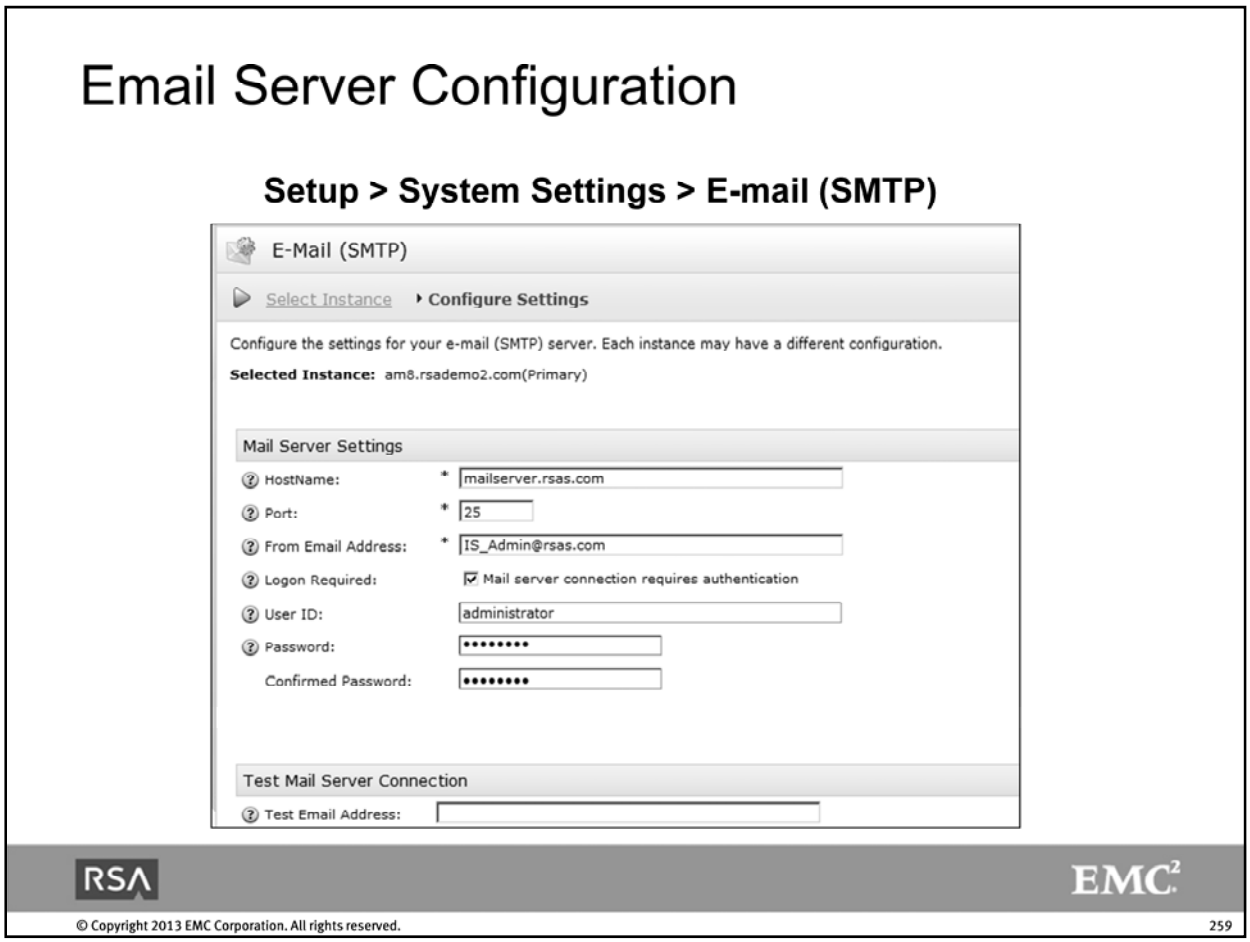
RSA EMC²

258

Security Domain configuration defines which security domains are available for enrolling users to add their accounts.

User-friendly names can also be configured for those domains.

The security domains that you make available for user enrollment determine which users administrators can manage, and limit the scope of the administrators' control by limiting the security domains to which they have access.



E-mail is sent automatically when users submit requests for enrollment, tokens, the on-demand tokencode service, or user group membership. The recipients of the e-mail can be users, approvers, distributors, Super Admins, and workflow participants in the parent security domain.

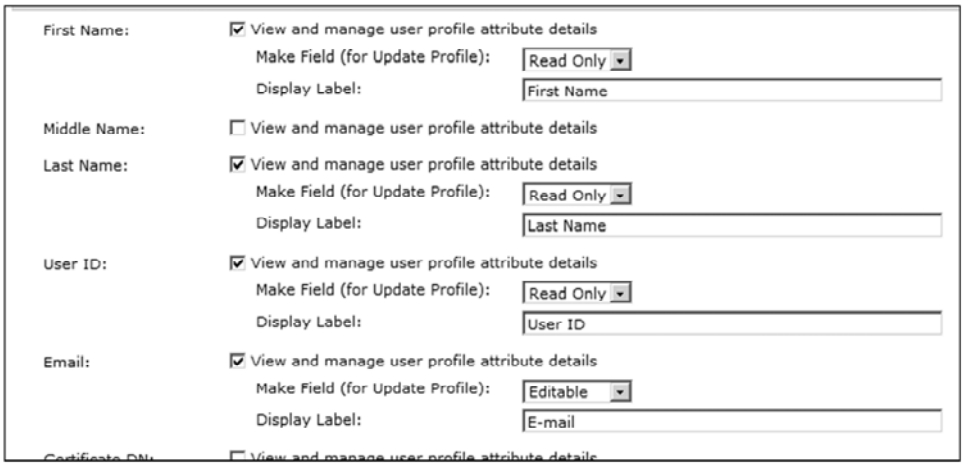
E-mail is sent to workflow participants when:

- Users submit requests that require approval or that require distributors to take action.
- Requests fail or cannot be processed.
- Approvers approve, reject, or cancel requests.

E-mail configurations are made from the Setup menu of the Security Console.

User Profile Configuration

- Customize what fields are Required, Editable, Read-only or Hidden (**Make Field** setting)
- Similar Option to provide display label for each entry field



RSA EMC²

260

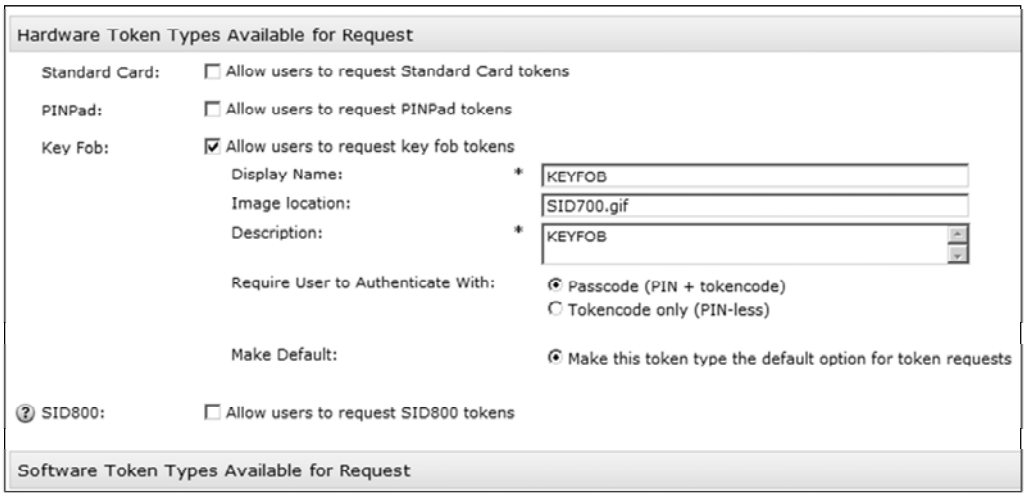
You can customize what fields are Required, Editable, Read-only or Hidden (using the “Make Field” setting)

You also have the option to provide helpful text for each entry field and friendly label field.

Mandatory attributes have only a ‘Required’ Make-Field option (Last Name, User Id, Email, Password and Confirm Password).

Authenticator Configuration

- Define which token types can be requested
- Define emergency access parameters for different conditions (Token lost/broken, temporarily unavailable, etc)



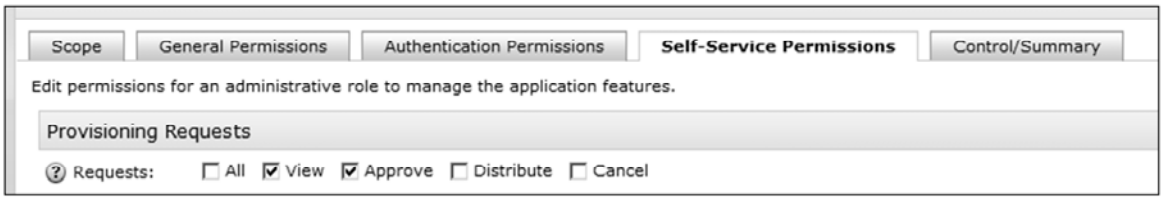
Users can request SecurID tokens using the Self-Service Console. You must configure the types of tokens that you want to make available to users. You can also set a default token type for all token requests.

The Token Configuration defines how tokens are distributed and how you want users to request tokens.

For example, what information they need to provide when requesting tokens and what types of tokens they can request.

Roles, Permissions

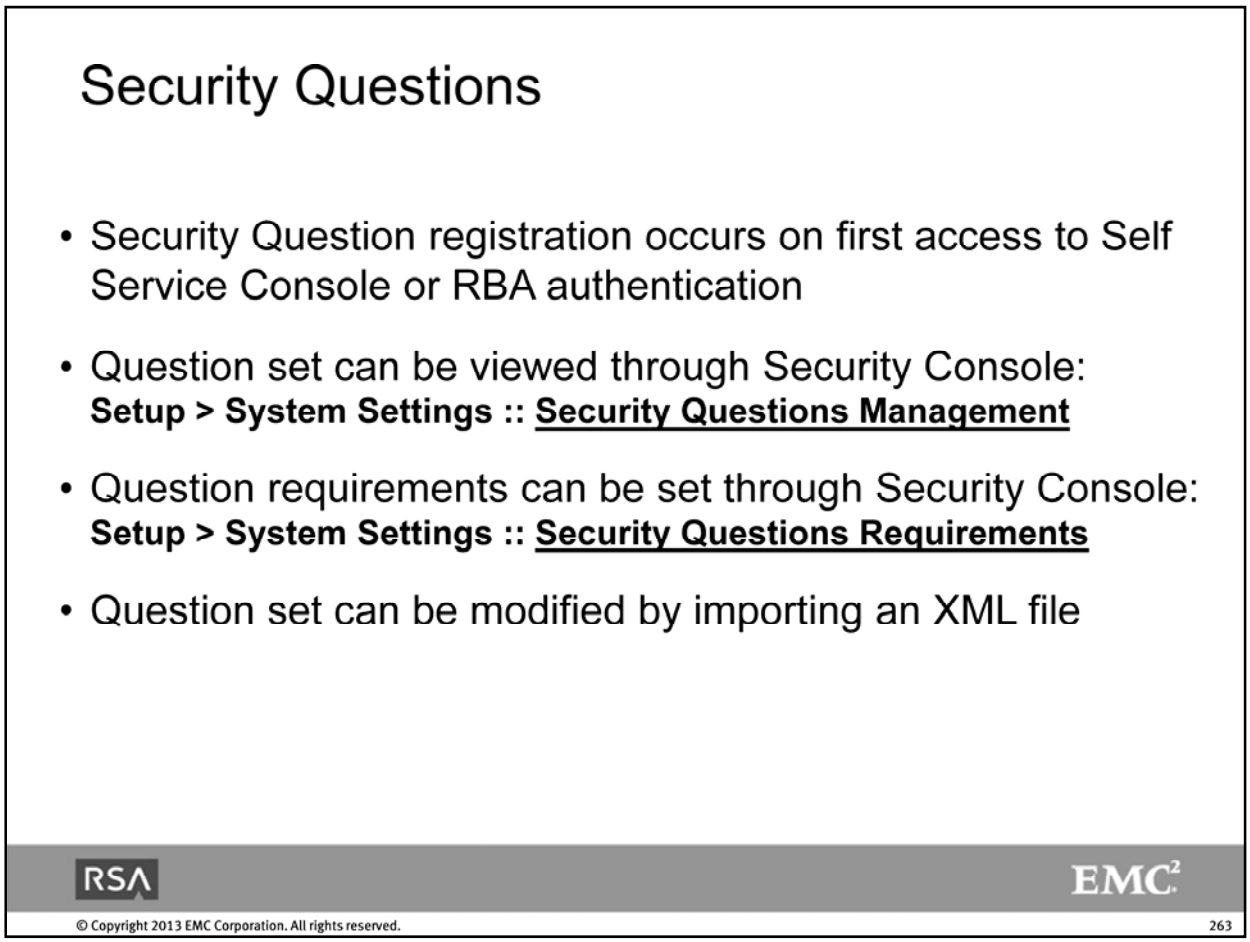
- Set permission(s) for Approver/Distributor
  - [Perform]All, View, Approve, Distribute, Cancel



RSA EMC²

262

You can set permissions for the Approver: to View, Approve, Distribute, Cancel, or All functions.



Security question registration occurs on a user's first access to the Self-service Console.

Security question data cannot come from an Identity Source.

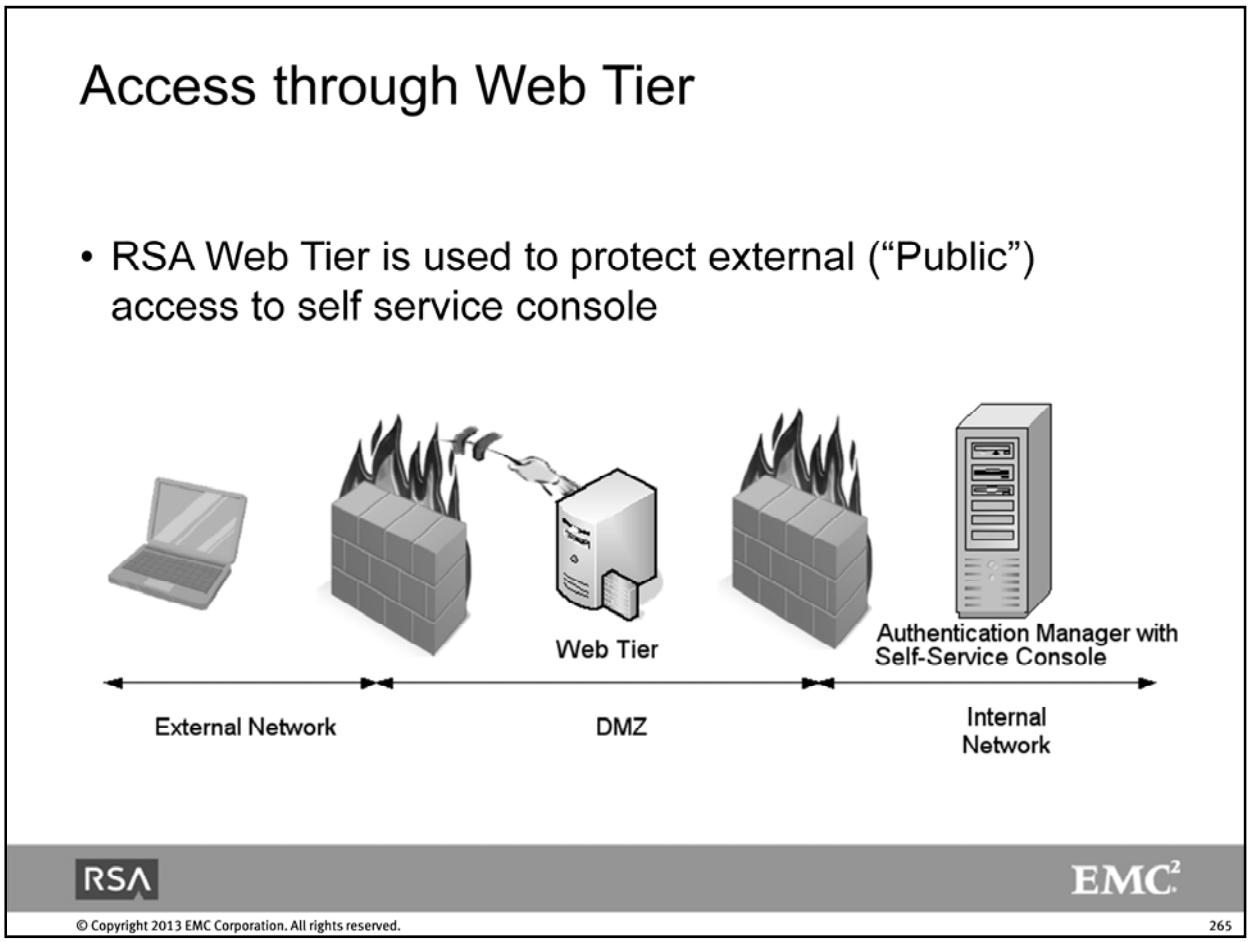
Security Questions *(cont'd)*

- Imported file replaces all questions for that language set – even if only one question is changed
- Import file is the following general format:

```
<?xml version="1.0" encoding="UTF-8"?>
  <SECURITY_QUESTIONS>
    <LANGUAGE id="en_US">
      <QUESTION>first question</QUESTION>
      <QUESTION>second question</QUESTION>
      <QUESTION>third question</QUESTION>
    </LANGUAGE>
  </SECURITY_QUESTIONS>
```

- Sample XML file is contained in RSA Authentication Manager Distribution Files

RSA EMC²



It may be desirable to make access to self-service available to users outside of a protected network. The RSA Web Tier can be configured to both allow external access to self service console while maintaining network protection.

Exercise: Self-Service

- Configure Self-service for user enrollment
- Enroll for, approve, and retrieve token

VARIABLES:
<mail_server_host>
<mail_server_port>
<email_username>

RSA EMC²

Troubleshooting

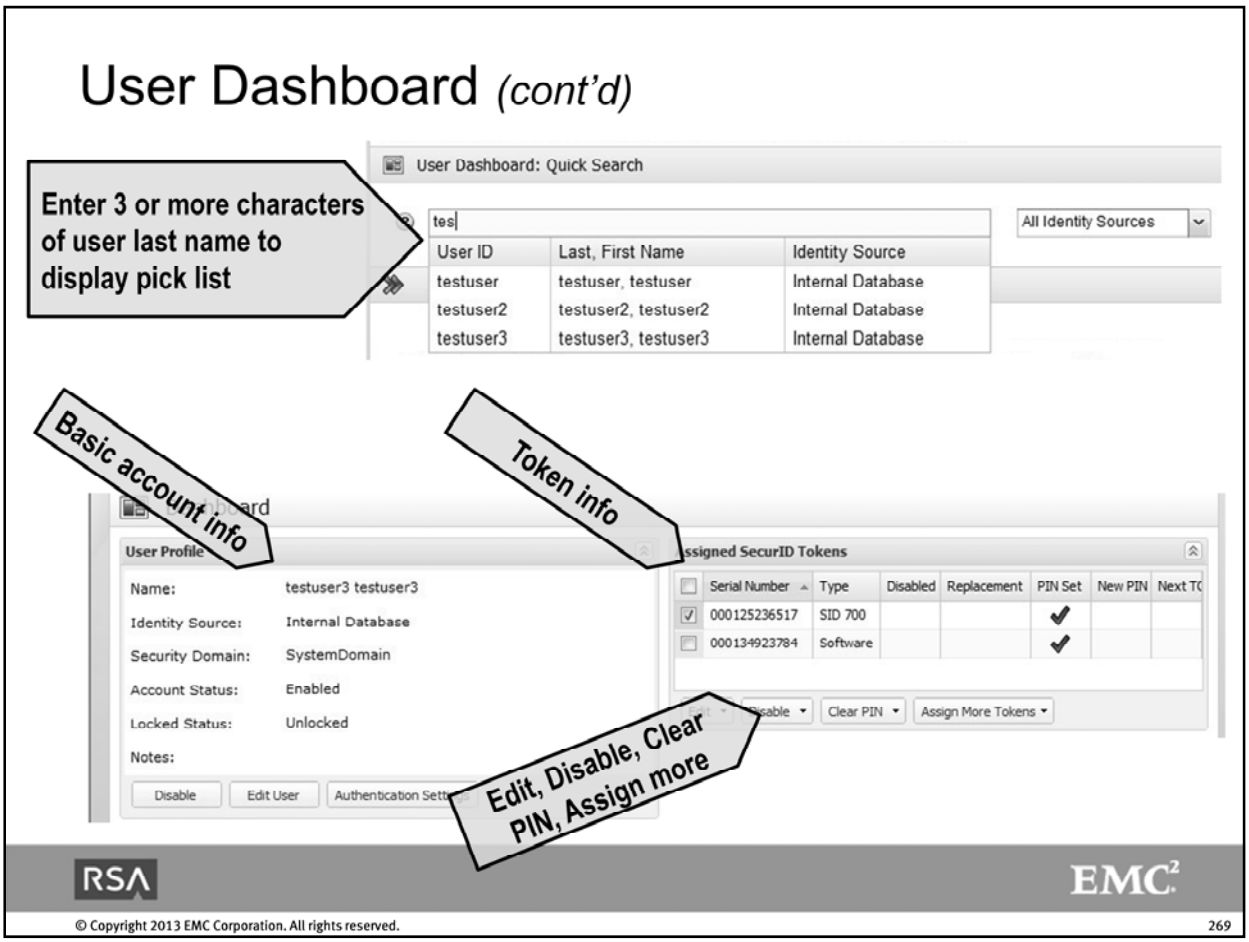
Unit 15

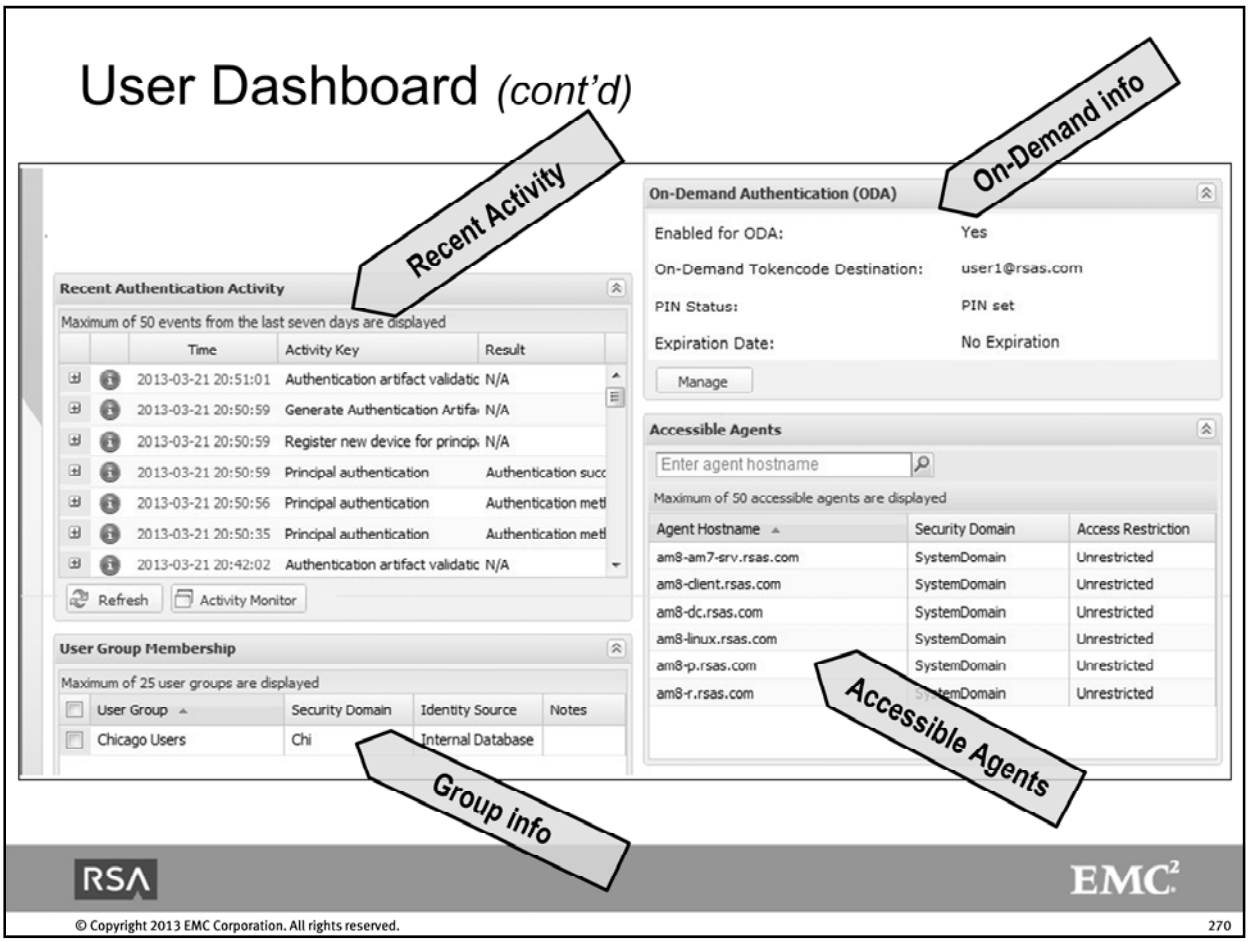
RSA EMC²

267

User Dashboard

- Rapid access to a specific user via “Quick Search”
- Provides view into:
  - User status (locked/unlocked, enabled/disabled)
  - Identity Source & Security Domain
  - Last 50 events / last 7 days’ activity
  - Assigned Tokens & On-Demand settings
  - Group memberships
  - Accessible Agents

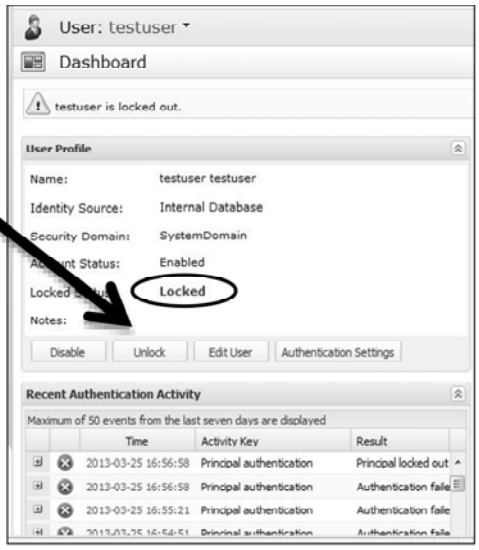

User Dashboard (cont'd)
Enter 3 or more characters of user last name to display pick list
User Dashboard: Quick Search
tes
All Identity Sources
User ID
Last, First Name
Identity Source
testuser
testuser, testuser
Internal Database
testuser2
testuser2, testuser2
Internal Database
testuser3
testuser3, testuser3
Internal Database
Basic account info
Token info
User Profile
Name:
testuser3 testuser3
Identity Source:
Internal Database
Security Domain:
SystemDomain
Account Status:
Enabled
Locked Status:
Unlocked
Notes:
Disable
Edit User
Serial Number
Type
Disabled
Replacement
PIN Set
New PIN
000125236517
SID 700
000134923784
Software
Clear PIN
Assign More Tokens
Edit, Disable, Clear PIN, Assign more
RSA
EMC²
© Copyright 2013 EMC Corporation. All rights reserved.
269

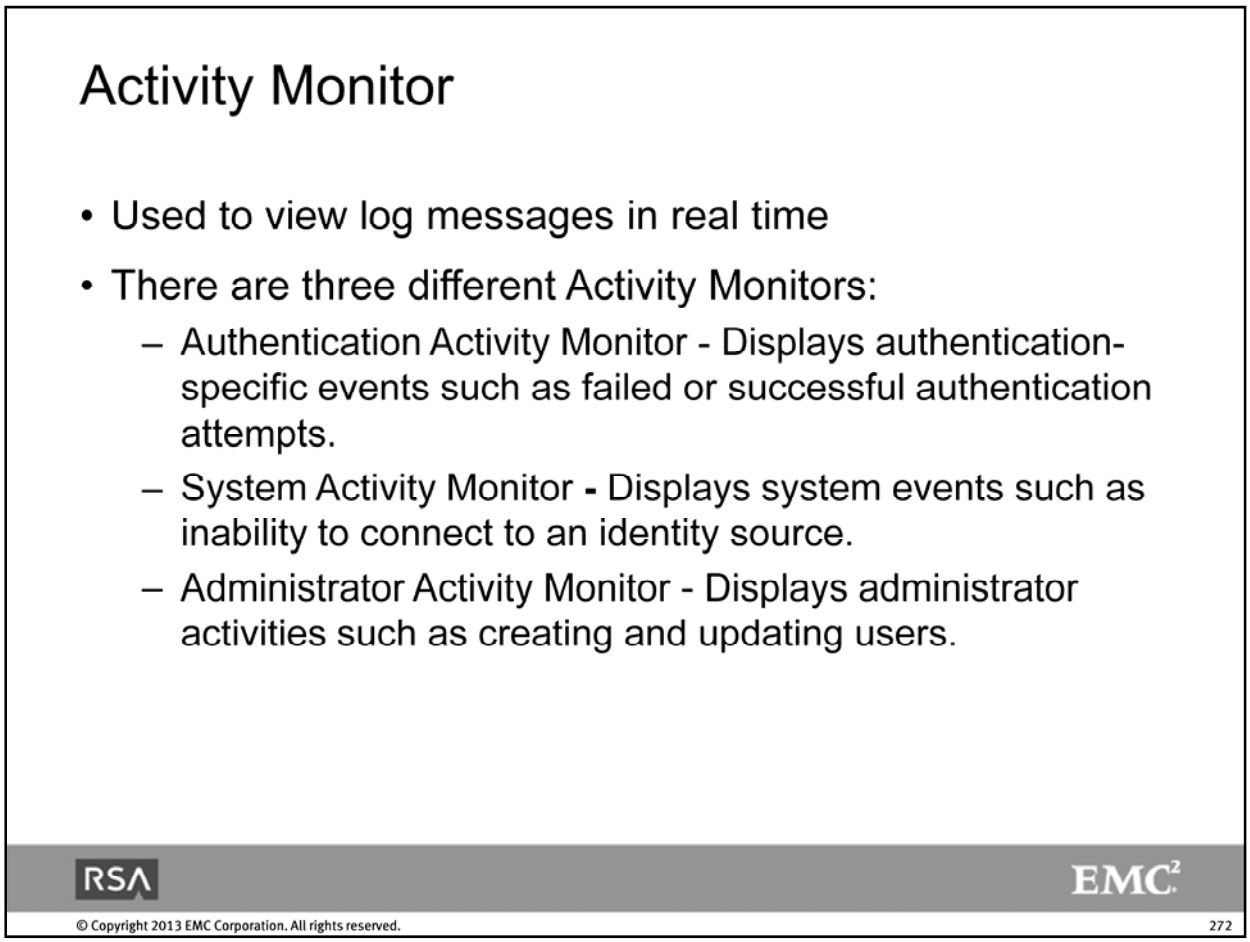

User Dashboard (cont'd)
Recent Activity
On-Demand info
Recent Authentication Activity
Maximum of 50 events from the last seven days are displayed
Time
Activity Key
Result
2013-03-21 20:51:01 Authentication artifact validatic N/A
2013-03-21 20:50:59 Generate Authentication Artifa N/A
2013-03-21 20:50:59 Register new device for princip N/A
2013-03-21 20:50:59 Principal authentication Authentication succ
2013-03-21 20:50:56 Principal authentication Authentication metl
2013-03-21 20:50:35 Principal authentication Authentication metl
2013-03-21 20:42:02 Authentication artifact validatic N/A
Refresh
Activity Monitor
On-Demand Authentication (ODA)
Enabled for ODA: Yes
On-Demand Tokencode Destination: user1@rsas.com
PIN Status: PIN set
Expiration Date: No Expiration
Manage
Accessible Agents
Enter agent hostname
Maximum of 50 accessible agents are displayed
Agent Hostname
Security Domain
Access Restriction
am8-am7-srv.rsas.com SystemDomain Unrestricted
am8-client.rsas.com SystemDomain Unrestricted
am8-dc.rsas.com SystemDomain Unrestricted
am8-linux.rsas.com SystemDomain Unrestricted
am8-p.rsas.com SystemDomain Unrestricted
am8-r.rsas.com SystemDomain Unrestricted
User Group Membership
Maximum of 25 user groups are displayed
User Group
Security Domain
Identity Source
Notes
Chicago Users Chi Internal Database
Group info
Accessible Agents
RSA
EMC²
© Copyright 2013 EMC Corporation. All rights reserved.
270

User Dashboard *(cont'd)*

If user is locked out, Dashboard displays 'Unlock' button

Recent activity can be expanded to view detail





The Activity Monitors are used to view log messages in real time as messages are logged to the Authentication Manager database.

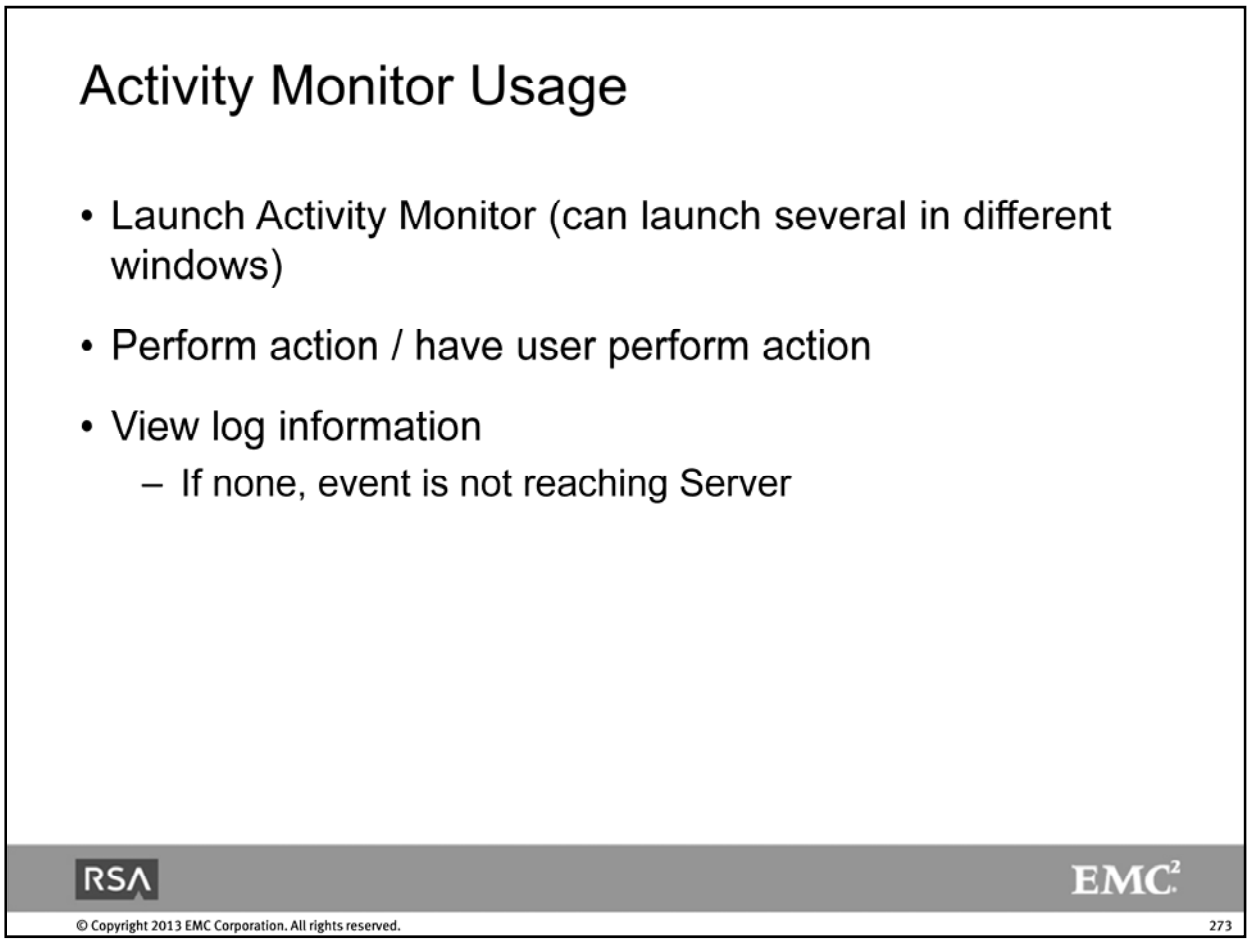
There are three different Activity Monitors:

- Authentication Activity Monitor

  Displays authentication-specific events such as successful authentication attempts.
- System Activity Monitor

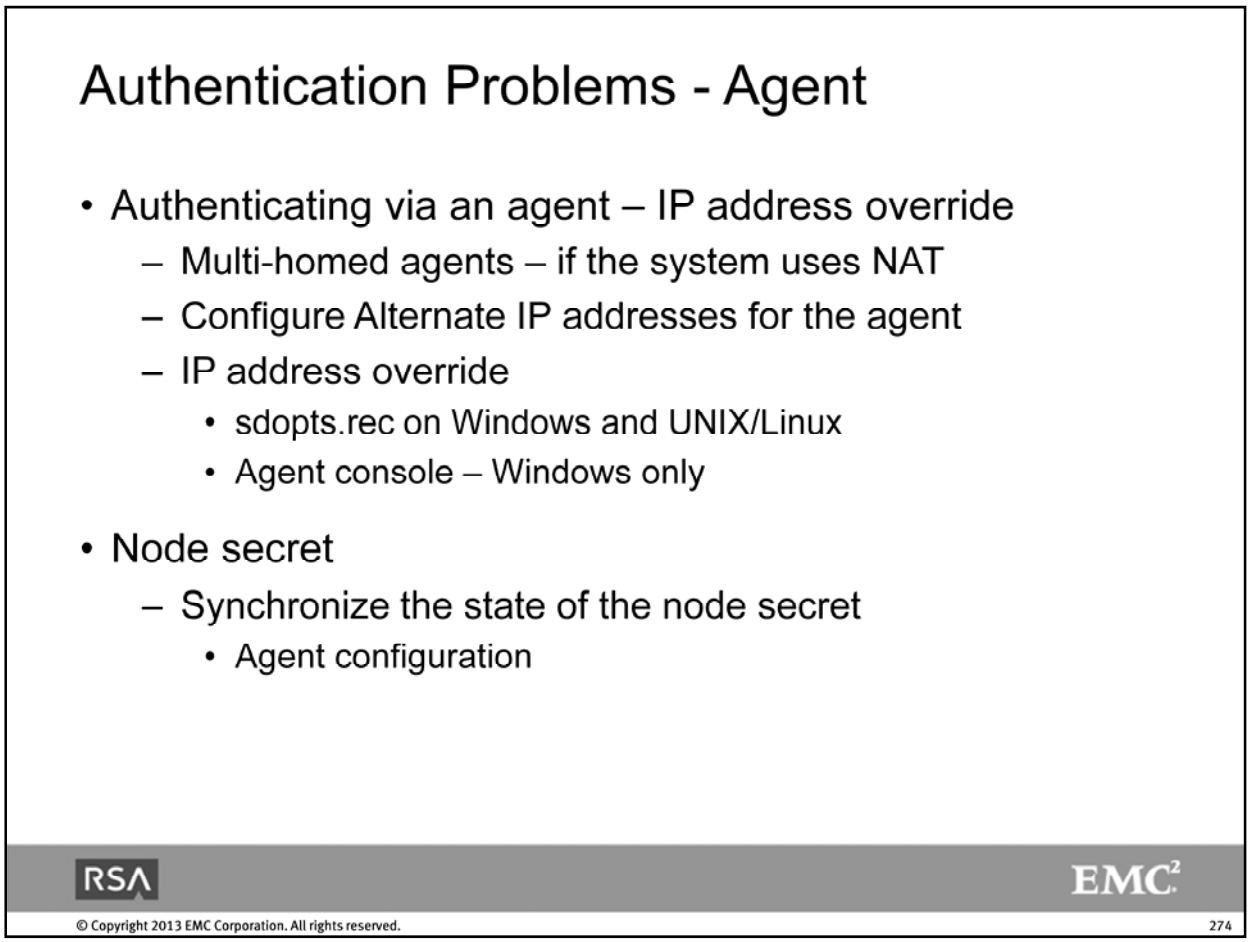
  Displays system events such as inability to connect to an identity source.
- Administrator Activity Monitor

  Displays administrator activities such as creating and updating users.



The process to use in troubleshooting when using the Activity Monitor follows this general approach:

1. Launch the Activity Monitor (you can launch several in different windows)
2. Perform the action you are trying to follow or troubleshoot (or have the user perform the action they are having trouble with - for example, performing an authentication.
3. View the log information as it appears in the Monitor window.
4. If there is none, conclude that the event is not reaching the Server and begin troubleshooting connection or Agent components.



If an Agent is not communicating with an authentication server, the sent/received IP address between the Agent and server may be a problem. The Agent will send data packets to the server with the packet containing the source IP address. If the server does not recognize the address based on the Agent record in the database, it will ignore or reject the packet.

An IP address override can be configured in some Agents to establish the address that will be used in server communication - this is applicable when an Agent is a multi-homed device or if the system uses NAT (Network Address Translation).

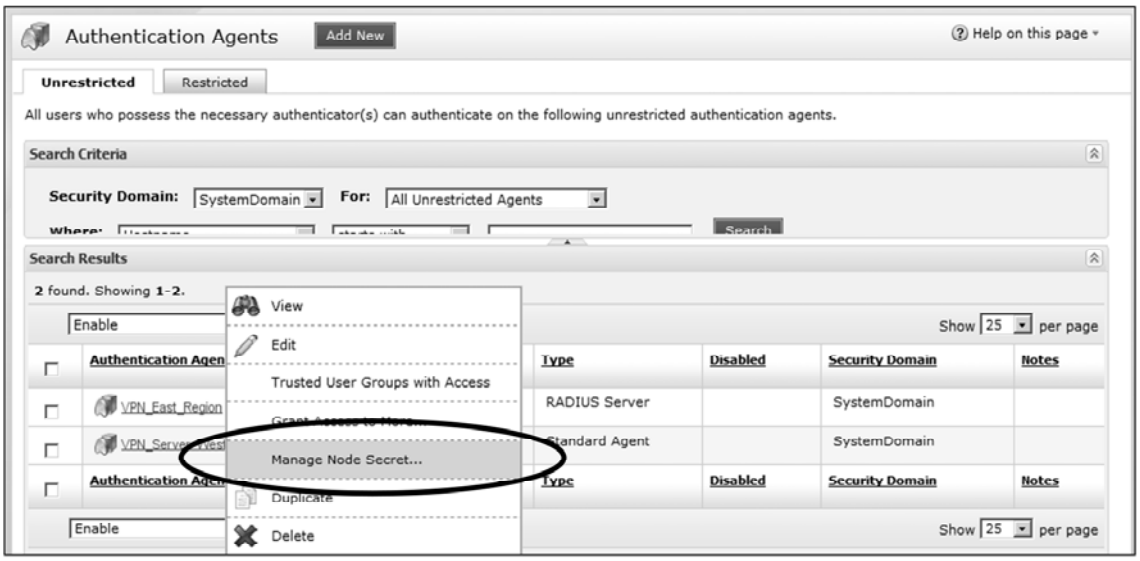
Alternate IP addresses can be configured for the Agent as part of the Agent record so that the server will recognize Agent communications.

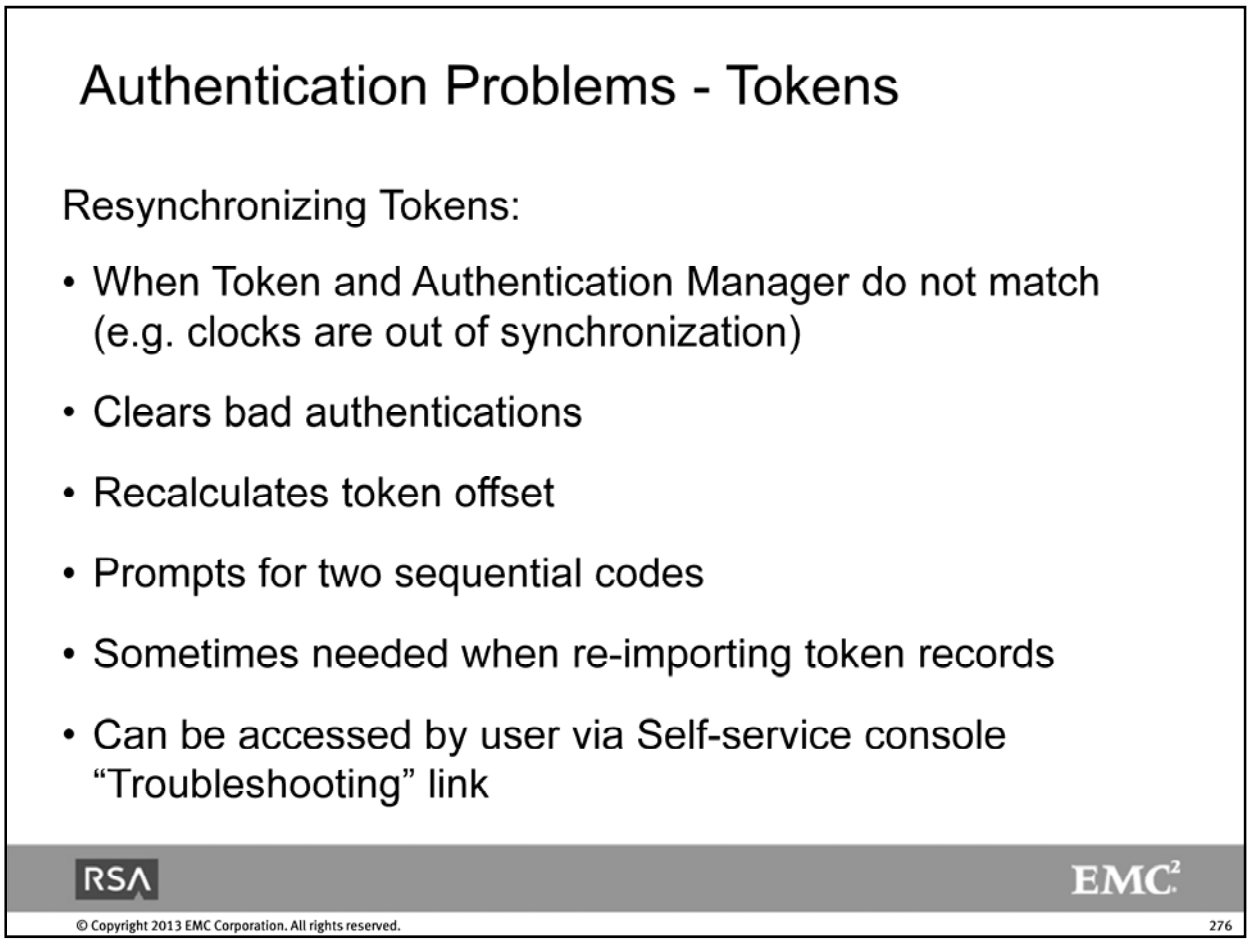
An sdopts.rec file can be configured for Agents if establishing a destination from the Agent to the Server is a problem - the sdopts.rec file can specify the target address of the server for the Agent.

If there is a Node Secret mis-match between the Agent and Server, communications will not be properly encrypted/decrypted. Synchronizing the state of the node secret may correct this problem - delete or clear the node secret at the Agent (depending on the Agent type) and clear the node secret in the Agent record in the server database. This will re-establish the common default node secret at both ends of the communication link.

Authentication – Manage Node Secret

- Access Node Secret management from Agent screen





If a Token is out of synchronization, it may be resynchronized through the Security Console.

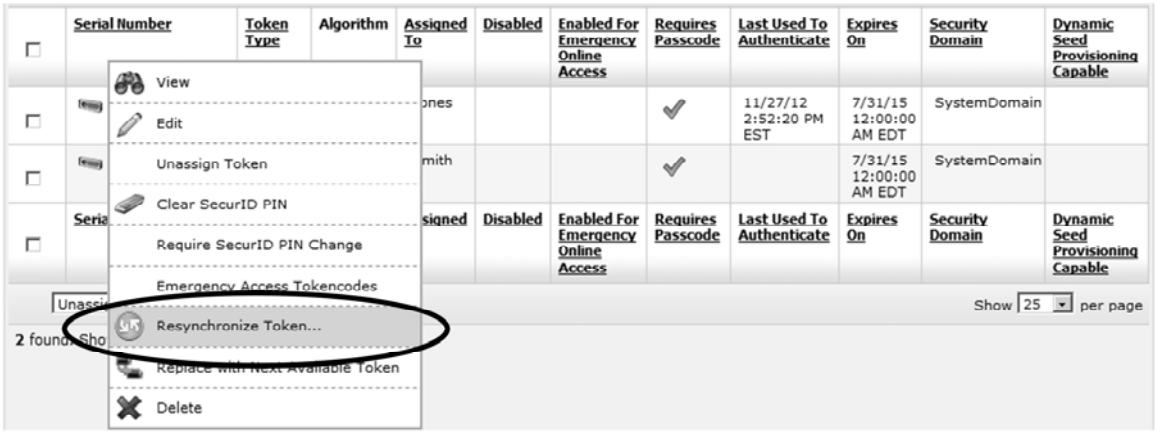
For Time based tokens resynchronization does the following:

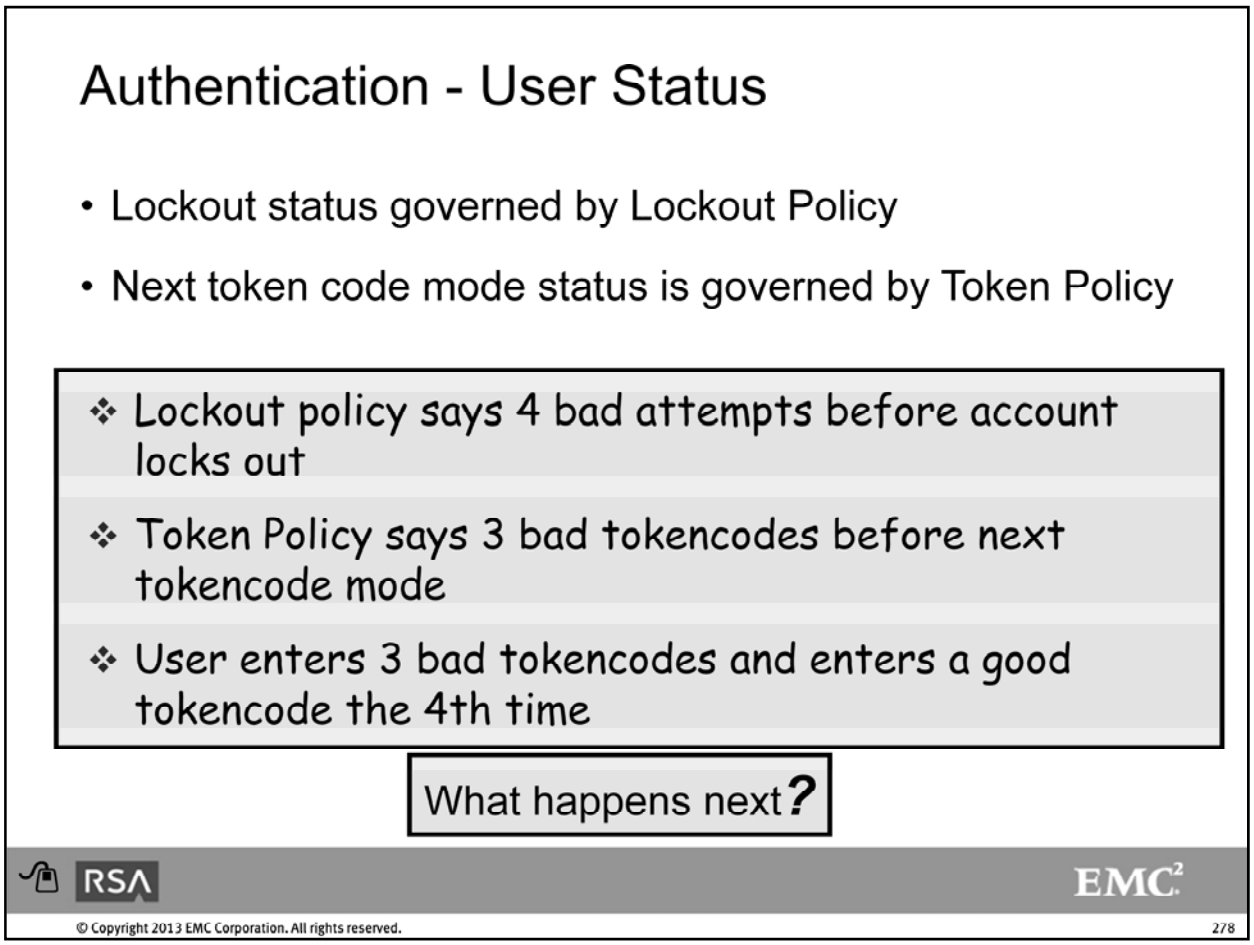
- Corrects a condition where the Token and Authentication Manager clocks do not match
- Clears bad authentications count
- Recalculates the Token offset

The process prompts for two sequential tokencodes - these are either supplied to the help desk buy a user calling in or, if the token is in hand prior to being re-deployed, are read directly from the Token.

Authentication – Resynchronizing Tokens

- Token resync is accessed through the token record (Manage Tokens screen)





A Locked Out user status is governed by the Lockout Policy. Similarly, the Next token code mode status is governed by the Token Policy.

Unlocking a user can be performed manually by an administrator or, if policy allows, the lockout is cleared after a set time period.

If a user synchronized with an LDAP directory is locked out in LDAP, the user must be unlocked through the LDAP directory (by the appropriate administrator).

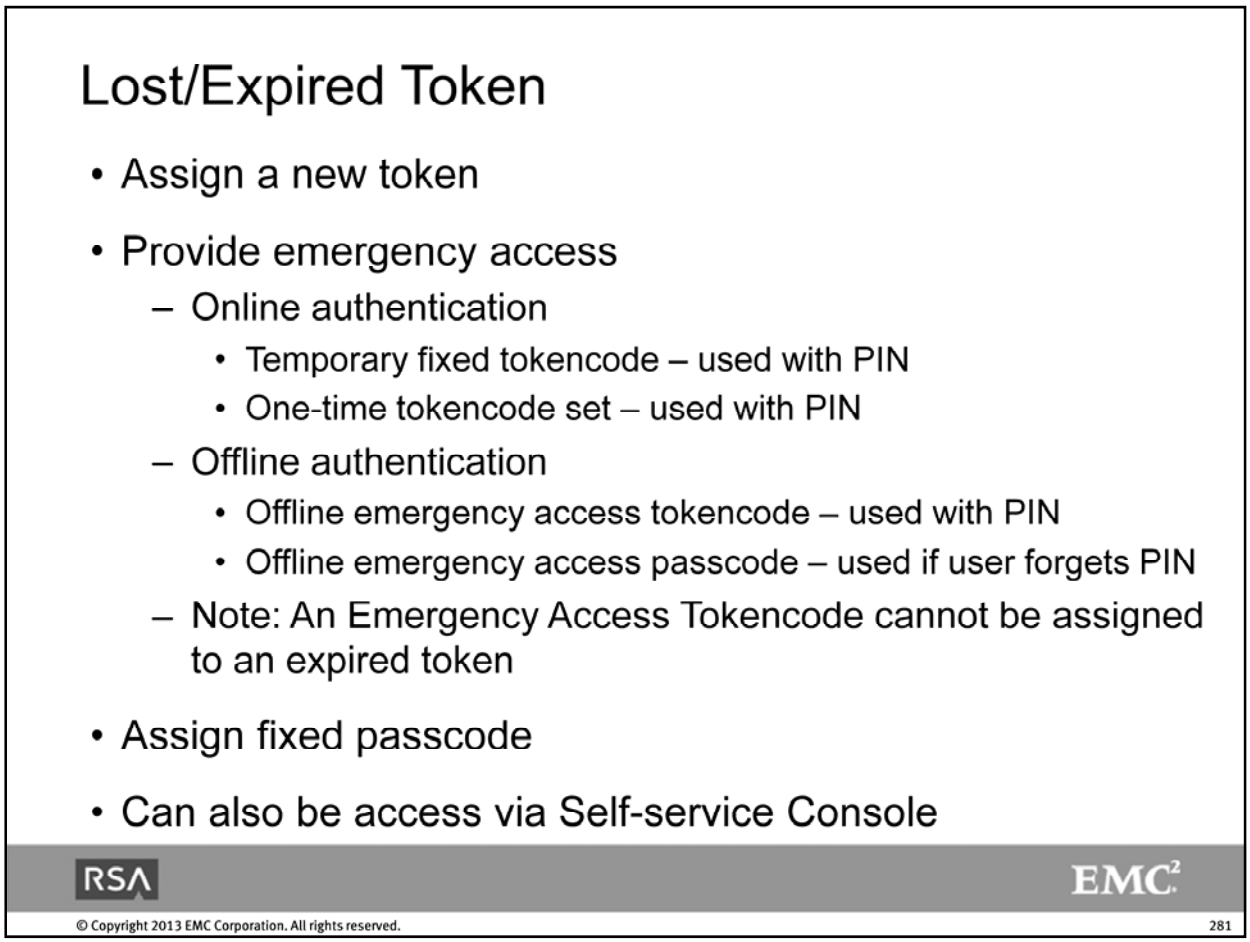
Authentication - User Status *(cont'd)*

- Unlocking user
  - User locked out in Authentication Manager
    - Wait until lockout expires (if configured) –or–
    - Admin intervention
  - User locked out in LDAP
    - Unlock the user in LDAP (Admin intervention)

RSA EMC²

Clearing Disabled status, Lockout, Security Questions

- Access through User's context menu (Authentication Settings)



If a user has lost their token or it has expired unexpectedly, several steps may be taken.

- The user can be assigned a new token.
- The user can be assigned Fixed Passcode.
- An administrator can provide emergency access:
  – For Online authentication, there is an option of supplying:
  - A Temporary fixed tokencode – used with the user's established PIN
  - A One-time tokencode set – used with the user's established PIN
  – For Offline authentication, there is an option of supplying:
  - An Offline emergency access tokencode – used with the user's established PIN
  - An Offline emergency access passcode – used if the user forgets their PIN

Note: An Emergency Access Tokencode cannot be assigned to an expired token

Exercise: Troubleshooting Authentication Issues

- Access menus and tools for end user troubleshooting and assistance
  - User Dashboard
  - Activity Monitor
  - Resynchronize Token
  - Emergency Codes
  - End user Self-Service

Approx. 20 minutes

RSA EMC²