NSS

Access Control

Access Control Terminology

Identification and Authentication

- Identification: unproven assertion of identity
  - “My name is…”
  - Userid
- Authentication: proven assertion of identity (BETTER)
  - Userid and password
  - Userid and PIN
  - Biometric

Authentication Methods

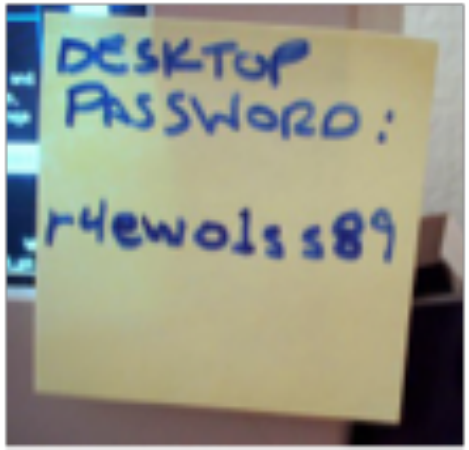
- What the user *knows*
  - Userid and password
  - Userid and PIN
- What the user *has*
  - Smart card
  - Token
- What the user *is*
  - Biometrics (fingerprint, handwriting, voice, etc

How Information Systems Authenticate Users

- Request userid and password
  - Hash password
  - Retrieve stored userid and hashed password
  - Compare
- Make a function call to a network based authentication service

How a User Should Treat Userids and Passwords

- Keep a secret
- Do not share with others
- Do not leave written down where someone else can find it
- Store in an encrypted file or vault



Password Hashes

Cain, Cracker top tab, right-click empty space, Add to List

LM hash is weak, no longer used in Win 7

NT hash is stronger, but not salted

k | Sniffer | Cracker | Traceroute | CCDU | Wireless | Query

User Name	LM Password	< 8	NT Password	LM Hash	NT Hash
Administrator	* empty *	*	* empty *	AAD3B435B5140...	31D6CFE0D16AE931B73C59D7E0C089C0
Guest	* empty *	*	* empty *	AAD3B435B5140...	31D6CFE0D16AE931B73C59D7E0C089C0
HomeGroupUser$	* empty *	*		AAD3B435B5140...	DC67B8EE5E3D3871B54CB574A259651A
Sam	* empty *	*		AAD3B435B5140...	E0269B792092CF3E924E080EFB029908
__vmware_user__	* empty *	*		AAD3B435B5140...	31D653D2A4BE64036586BF7B2C81C965

Strong Authentication

- Traditional userid + password authentication has known weaknesses
  - Easily guessed passwords
  - Disclosed or shared passwords
- Stronger types of authentication available, usually referred to as “strong authentication”
  - Token
  - Certificate
  - Biometrics

Two Factor Authentication

- First factor: what user knows
- Second factor: what user has
  - Password token
  - USB key
  - Digital certificate
  - Smart card
- Without the second factor, user cannot log in
- Defeats password guessing / cracking

Biometric Authentication

- Stronger than userid + password
- Stronger than two-factor?
  - Can be hacked

Biometric Authentication

- Measures a part of user’s body
  - Fingerprint
  - Iris scan
  - Signature
  - Voice
  - Etc.

Authentication Issues

- Password quality
- Consistency of user credentials across multiple environments
- Too many userids and passwords
- Handling password resets
- Dealing with compromised passwords
- Staff terminations

Access Control Technologies

Centralized management of access controls

- LDAP
  - Active Directory, Microsoft's LDAP
- RADIUS
  - Diameter, upgrade of RADIUS
- Kerberos
    - Uses Tickets

Single Sign-On (SSO)

- Authenticate once, access many information systems without having to
- re-authenticate into each
- Centralized session management
- Often the “holy grail” for identity management
  - Harder in practice to achieve – integration issues

Reduced Sign-On

- Like single sign-on (SSO), single credential for many systems
- But… no inter-system session management
- User must log into each system separately, but they all use the same userid and password

Weakness of SSO and RSO

- Weakness: intruder can access all systems if password is compromised
- Best to combine with two-factor / strong authentication

Conceptual Concepts

- Principles of access control
- Types of controls
- Categories of controls

Principles of Access Control

- Separation of duties
  - No single individual should be allowed to perform high-value or sensitive tasks on their own
    - Financial transactions
    - Software changes
    - User account creation / changes

Principles of Access Control

- Least privilege
  - Persons should have access to only the functions / data that they require to perform their stated duties
- Server applications
  - Don't run as root
- User permissions on File Servers
  - Don't give access to others' files
- Workstations
  - User Account Control

Principles of Access Control

Defense in depth

- Use of multiple controls to protect an asset
- Heterogeneous controls preferred
  - If one type fails, the other remains
  - If one type is attacked, the other remains

Examples

- Nested firewalls
- Anti-virus on workstations, file servers, e-mail servers

Controls

Deterrent Controls

- A purely deterrent control does not prevent or even record events
  - Warning banners/Signs
  - Guards, guard dogs (may be preventive if they are real)
  - Razor wire

Preventive Controls

- Block or control specific events
  - Firewalls
  - Anti-virus software
  - Encryption
  - Key card systems
  - Bollards stop cars (as shown)

Corrective Controls

- Post-event controls to prevent recurrence
- “Corrective” refers to when it is implemented
  - Can be preventive, detective, deterrent, administrative
- Examples (if implemented after an incident)
  - Spam filter
  - Anti-virus on e-mail server
  - WPA Wi-Fi encryption

Recovery Controls

- Post-incident controls to recover systems
- Examples
  - System restoration
  - Database restoration

Compensating Controls

- Control that is introduced that compensates for the absence or failure of a control
- “Compensating” refers to why it is implemented
- Can be detective, preventive, deterrent, administrative
- Examples
  - Daily monitoring of anti-virus console
  - Monthly review of administrative logins
  - Web Application Firewall used to protect buggy application

Testing Access Control

Testing Access Controls

- Access controls are the primary defense that protect assets
- Testing helps to verify whether they are working properly
- Types of tests
  - Penetration tests
  - Application vulnerability tests
  - Code reviews

Penetration Testing

- Automatic scans to discover vulnerabilities
  - Scan TCP/IP for open ports, discover active “listeners”
  - Potential vulnerabilities in open services
  - Test operating system, middleware, server, network device features
  - Missing patches
- Example tools: Nessus, Nikto, SAINT, Superscan, Retina, ISS, Microsoft Baseline Security Analyzer

Application Vulnerability Testing

- Discover vulnerabilities in an application
- Automated tools and manual tools
- Example vulnerabilities
  - Cross-site scripting, injection flaws, malicious file execution, broken authentication, broken session management, information leakage, insecure use of encryption, and many more

Audit Log Analysis

- Regular examination of audit and event logs
- Detect unwanted events
  - Attempted break-ins
  - System malfunctions
  - Account abuse, such as credential sharing
- Audit log protection
  - Write-once media
  - Centralized audit logs