Chapter 2

Sunday, July 30, 2023 12:39 PM

Users claim identity w/ username, prove with password, and granted access based on identity

Exploring Authentication Management

authentication - proves identity w/ some form of credentials
ex: uname/password

identification - when users claim their identity

at least two entities know the credentials
ex: user & authenticator
user presents, authenticator verifies

cant have any type of access control w/out authentication
if everyone anonymous, then everyone has all access

systems use authentication too: servers, processes, workstations...

mutual authentication - both parties authenticate each other

Comparing Identification and AtA

authentication, authorization, and accounting work with identification to create access management system

identification → authentication → ...

authorization - permission to access based on proven identity

resources
principle of least privilege

accounting - tracks user activity and record in logs
ex.. audit logs track activity and admins use to create audit trail - recreate events that preceded an incident

effective access controls rely on strong authentication, if users can bypass them authorization and accounting are ineffective

Comparing authentication factors

types or factors for authentication
1 factor for basic, 2 for secure

3 factors of authentication

- Something you know (pword, PIN, ...)
- Something you have (smart card, phone)
- Something you are (biometrics)

Something you know

typically a shared secret such as a pword, static code, or PIN

least secure form of authentication

static codes - or static pwords that stay the same for a long time

NIST SP 800-63B "Digital identity guidelines" says pwords are memorized secrets that are easy to remember hard to guess

Current recommendations

- hash all pwords
- MFA
- [illegible] pword resets

- Don't require mandatory
- at least 8 chars
- Check for common pwords
- don't use same work password anywhere else
- allow special chars but don't require

Password complexity

one method is to require pwords to be complex and strong

Strong pword is good length, no dict words, words in any part of user's name, combines at least 3:

- uppercase
- lowercase
- #s
- special char

Password expiration

when users need to change pword

"maximum password age"

Password history and Password Reuse

password history system - remembers past pwords and prevents them from being used

common to remember past 24 pwords

minimum password age - how long a password must last until changing again

Password vaults

password manager - single source designed

to keep most

keeps pwords in encrypted format

Password keys

used to reset passwords on systems

bootable optical disk or bootable USB Drive

after rebooting system to device, they can recover or reset all user and admin pwords

people who forgot pwords or forensics who need access w/o knowing pword

also used by attackers who stole pc

many of these are free to download on internet, but could be malware

knowledge based authentication

knowledge based authentication KBA

two types: static & dynamic

Static KBA

typically when you've forgot pword to verify identity

ex: security questions

Dynamic KBA

identifies people w/o account

used for high-risk transactions

queries public and private data

source

then makes multiple choice questions that only user would know

ex: which addresses have you lived in?

Which is closest to mortgage payment?

How much is car payment?

Implementing Account Lockout Policies

lockout policies - prevent users from guessing the password

account lockout threshold - max # of times a user can enter wrong pword

account lockout duration - how long account remains locked

Changing default passwords

many systems start w/ default pwords

also includes changing default name

usually defaults to "admin"

can even setup a dummy account called admin

Training users about password behaviors

training on:

- creating strong pwords
- not using same pwords in other places
- changing pword

. never share

Something you have

Something you can physically hold
Smart cards, common access cards, hardware tokens

Smart card authentication

Smart cards - credit card sized cards that have embedded microchip and certificate
insert into smart card reader

certificate-based authentication
certificates - digital files that support cryptography for increased security
also can be used w/ digital signatures and data encryption

embedded certificate - holds user's private key and matched w/ public key

public key infrastructure **PKI** - issues and manages certificates

smart cards often used w/ other authentication like PIN or password

Token key

token key - aka token fob or token; device about size of car key
LCD display that shows # that changes periodically

hardware token,

token is synced to server that knows what # is at any moment

is a one-time use, rolling password

even if attackers knows it, not useful for very long

RSA sells RSA secure ID, a popular token

HOTP and TOTP

hash-based message authentication code HMAC - uses hash function and cryptographic key for many different cryptographic functions

HMAC-based one-time password HOTP

open standard to create one time pwords

Similar to tokens

algorithm combines secret key and incrementing counter, and uses HMAC to create hash of result

converts result into HOTP value of 6-8 digits

Request new HOTP # using token or software

expires after use, can't use again, nobody else can use

HOTP password remains valid until used, so if you generate then forget to use it, it is still

valid

time-based one-time password TOTP
uses timestamp instead of counter
typically expire after 30 sec

HOTP and TOTP are inexpensive

Authentication applications

many software apps use these algos to create software tokens

Some have created TOTP or HOTP apps but modified with proprietary software like google authn

Two-step verification

two-factor authentication
adds extra layer of security to account
most rely on having phone

Short message service (SMS) to send PINs

NIST SP-800-63B discourages SMS
texts typically displayed on lockscreen

Option to receive phonecall as well

push notifications - send messages to another device

think steam authn

Something you are

biometrics for authn most difficult

biometric
strongest form of auth; most
to falsify

Biometric Methods

two step: enroll with auth system
using id and fingerprint

types of biometrics

- fingerprint - laptops, phones, USB
  can store multiple fingerprints
  law enforcement use for identification
  not auth
- Vein matching - id users using
  near-infrared light to view veins
  mostly in palm for more veins
  many healthcare use to id patients
- retina - scan 1 or both eyes to see
  blood vessel patterns in back of eye
  some people object to because they
  can see medical issues
- iris - use camera tech to capture
  patterns around pupil
  used in passport free border
  crossings
- facial recognition - size of face,
  size, shape, position of eyes, nose,
  mouth, cheekbones, jaw
- voice - id who is speaking with
  acoustic features
- gait analysis - id people based
  walk

J
ou way they
how their feet hit and leave ground

formal enrollment not needed, some places are taking facial recog without knowing

iris and retina are strongest mentioned
iris used over retina because retina too intrusive

facial recog and gait analysis can bypass enrollment by using for identification instead of auth

Biometric Efficacy Rates

biometric efficacy rate - refers to the performance of system under ideal conditions

false acceptance - register unknown user as accepted user
false acceptance rate FAR

false rejection - incorrectly rejects known user
false rejection rate FRR

true acceptance
true rejection

biometric systems let you set sensitivity or threshold level where errors occur

\+ sens = less false matches
more false rejections

← SENS = more false ... less false rejections

crossover error rate CER - where FAR crosses FRR

lower CER = more accurate

Two factor and multifactor authentication

2FA often uses:
- smart card + PIN
- USB token + PIN
- hardware token + password

2 factors of same type ≠ 2FA

Authentication attributes

authentication attributes - identify users based on characteristics or traits

Somewhere you are

identifies a user's location

many auth systems use IP address for geolocation

gives details on country, region, state, city, and sometimes zip

can be used to identify impossible travel time

login in one place then right after login from another city

in an org, you can use pc name or MAC address of system for somewhere you are

Microsoft active directory domain ... ensure accounts ... only

can config
be able to be logged in
by certain PC

Something you can do
actions you can take such as
gestures on a touch screen
ex: picture pwords
select picture and add 3
gestures
tap specific places, draw lines,
circle item
use these gestures to login later

Something you Exhibit
you show or display
ex: a badge you wear at work
military gov can use common access
cards CAC or personal identity
verification PIV
use picture and personal info
also include smart card capabilities

Someone you know
Someone vouches for you
antivirus does this on web
ex: google w/ norton extension
will have verified sites that
are safe

web of trust - users create certs
themselves using decentralized
trust model over CA

B and gives cert

A trusts
B trusts cert
C trusts A and gives cert

if C gives B cert, B would trust because it trusts A which trusts C

Authentication Log Files

track successful and unsuccess logins

What happened - login success or fail

When - timestamp

Where - IP of pc name

Who - user account

Managing Accounts

account management - creation, management, disablement, and termination of accounts

when account is active, access control what user can do

also when, where and how users login

Credential policies and account types

credential policies - define login policies for different personnel, devices and accounts

personnel or end-user accounts

for regular employees

create accounts and give perms based on role

Local policy for all

basic credentials
employee like a pword policy

admin and root accounts - privileged accounts that have additional rights and privileges
admin = windows root = linux

Service accounts - some apps and services need to run under context of an account
admins can create regular user account for service like sql service and give required privileges

long complex pwords that dont expire

device accounts
pcs and other devices have accounts
ex: Microsoft active directory only allows users to logon to pcs in domain
these pcs have pc accounts and active directory manages pwords

third party accounts
external entities w/ access to network
ex: org has security apps that have admin access to network

Guest account
limited access to pc or network w/out account
ex: contractors that change frequently

Shared and generic account/credentials
regular user account that temp workers will share
discouraged for normal work
ex: temp agency send different people every day

Privileged Access Management
privileged access management PAM
apply more strict security controls for elevated privilege accounts
just-in-time - admins only have admin privileges when they need them
when they need them, account sends repest for elevated privileges
ex: add to privilege group then remove after time

capabilities
. allow auth users to access admin account without knowing passwd
. logging all elevated privileges usage
. change admin passwd

- limit time you can use account
- users to check out creds
- log all access of creds

PAM reduces opportunities for attackers to use admin privileges

Require admins to use two accounts

1 account for day-to-day same end-user privileges

1 account w/ elevated privileges

reduces chances of malware seizing admin account

reduce admin account risk for theft of pc or lunchtime attack

Prohibiting shared and Generic accounts

account management policies often discourage using shared/generic

instead each user has account

using shared accounts makes it hard to implement authorization, and accounting is less effective

note using a generic account for 1 user is fine, but when multiple its bad

Disablement policies

specify how to

disablement policy - specify how to manage accounts in different situations

ex: when someone leaves org

also, disable default accounts

disabling preferred over deletion, at least initially

deletion also deletes encryption and secret keys

ex: someone encrypts files w/ their account, OS uses their decryption key but will be gone after deletion

file could remain encrypted forever

account disablement policy

- terminated employee - ex-employees are disabled immediately
- leave of absence - disable account while away
- delete account - after certain amount of time

Time-based logins

users can only log on to pcs during specific times

if user works late, doesn't log them off but doesn't allow any new network connections

Account audits

account audits - look at rights and permissions assigned to users to enforce principle of least

Privilege

privilege creep or permission bloat -
when a user gains more and more permissions due to changing job requirements but unneeded ones aren't removed

Use role-based access controls with group-based privileges
need to change groups if employee changes from HR to Sales
permissions audit at least once a year

Remember
usage auditing records user activity in logs to determine behavior and make audit trail
permission auditing help ensure users only have access they need

Comparing Authentication Services
several other auth services to make sure unencrypted creds are not sent across networks

Single sign-on
log on once and access multiple systems without logging in multiple times
increase security because only need to know 1 pword
system typically creates an SSO secure token used during entire login

session
requires strong authentication because
1 login gives access to multiple
systems

Kerberos

Kerberos - network authentication
mechanism used within Windows
Active directory domains and some
Unix are known as realms
provides mutual authentication that
prevents on-path attacks
and uses tickets to prevent replay
attacks

Requirements to work properly

- method of issuing tickets used for authentication
  the key distribution center KDC - does
  complex process of issuing ticket
  granting tickets TGTs and other
  tickets

  tickets provide auth for users
  when they access resources like
  files on a server
  often called tokens, but not
  the same as key fob

- time synchronization - requires all
  systems to be time synced and
  within 5 min of each other
  clock that provides time sync is
  used to timestamp tickets
  to make sure they expire

prevents replay attacks because attacker only has limited time to impersonate user

- database of subjects or users - in windows this is active directory

remember

Kerberos is a network authentication protocol that uses database of subjects to issue timestamped tickets that expire

When user logs in with kerberos, the KDC issues the user a ticket-granting ticket which typically has lifespan of 10 hours

when users access resource, they present tgt as authentication and are issued ticket to access resource

if users stay on for long time key expires and requires relog

Kerberos uses symmetric key encryption to prevent unauthorized disclosure

SSO and a Federation

Some SSOs can connect auths from different env like OS or diff networks

common method is federated identity management system - often integrated as a federal database - offers central authentication in a non-

management

ex: nuclear site + school district dont want to join networks create federation of two networks so they can login with respective network accounts but still view shared resources

a federation requires a federated identity management system

creates federated identities - links user's creds from different networks or OS but treats as 1 identity

SAML

security assertion markup language

SAML - an Extensible Markup Language (XML) - based format used for SSO on web browsers

ex: two websites by two different orgs

normally separate logins but if orgs trust each other they can use SAML as federated identity management system

3 SAML roles

- principle - typically a user
  user logs in once
  if needed, principle requests identity from identity provider
- identity provider - IdP creates, maintains, and manages identity info for priciples
  individual orgs that

those are the two
principles are in

- Service provider - entity that provides services to principles
this is the secondary org that a user is trying to access resources from
hosts websites over web-based portal
queries IdP to verify valid creds before granting access

this process sends XML-based messages between systems

SAML and authorization

SSO is for identification and authentication
not for authorization

authorization is typically separate but many federation SSO systems, like SAML, include ability to transfer authorization data between systems

OAuth

OAuth - open standard for authorization many orgs use to provide secure access to protected resources

instead of accounts for each website you access, use same account through Google, Facebook, Paypal...

configure site

ex: ecommerce ... so to make
to use PayPal API
purchase they login w/ paypal

OAuth transfers data between
both

OAuth focuses on authorization,
not authentication

RFC 6749 "The OAuth 2.0 Authorization
Framework"

OpenID and OpenID Connection

OpenID - authentication standard
maintained by OpenID foundation

OpenID provider holds user creds
websites that support OpenID
prompt users to enter OpenID
sometimes OpenID prompts you to
give website other info

Not used that much but might
see OpenID Connection OIDC -
builds on OpenID for authorization
and uses OAuth 2.0 for
authentication

instead of authorization token, OIDC
uses JSON web token JWT

ex: have google account
"login with google"
app exchanges data w/ google
google exchanges JSON web token
with app
that website doesn't
credentials

benefit is ... need to store your cle...

Comparing Access Control schemes

access control that only authenticated and authorized users can access resources

access control schemes
- role-based access control
- rule-based access control
- discretionary access control DAC
- Mandatory access control MAC
- attribute based access control ABAC

Subjects - users or groups
sometimes a service

objects - files, folders, shares, printers that subjects access

Role based access control

role-BAC uses roles to manage rights and permissions

admins create rights and assign to roles rather than users

Using Roles Based on Jobs and Functions

different departments that each have different server hosting its files

create roles for each dept.
Accounting, Sales, IT ...

then grant roles access to respective servers

... is Microsoft project

another example

server - hosts multiple projects managed by different project managers

Microsoft project server roles

- admin - complete access and control over everything on server all projects on server
- executives - access data from any project held on server but can't modify server settings
- project managers - full control over their projects but not others
- team members - report on work pms assign to them but little access to outside of that

Documenting Roles with a Matrix

document role-based perms with a matrix listing all job titles and each role's privileges

Role	server priv	project priv
admin	All	All
exec	none	All
PM	none	all assigned projects
TM	none	assigned tasks only

Rob-Btc aka hierarchy-based or job-based

hierarchy-based - top level roles like admin have more perms than

lower

job, task, or function-based - roles based on job or tasks they need to perform

Establishing Access with Group-based Privileges

roles are often implemented as groups
Microsoft has built in groups like admin, backup operator, ...
often need to make custom groups
ex: split up backup operator into backup & restore

group-based privileges reduce admin workload for access management

Rule-Based Access Control

rule-BAC uses rules like in firewalls and routers but can be in apps too
routers and firewalls use rules within access control lists ACL - define traffic that device allow into network

most rules are static but rules in IPSs can change dynamically based on trends

can configure app rules too
ex: database rule to give someone permissions if work person is 000

Discretionary Access Control DAC

discretionary access control DAC - objects have owners that establish

access

many OS like Windows and unix-based use DAC

New technology File system (NTFS)
in windows
allows users and admins to restric access using permissions

File system permissions

NTFS permissions
- write - write to file
not delete
- Read
- Read and Execute
- Modify - view and change files like delete and add to folder
- full control - can do anything w/ file and its permissions

implicit deny - denies access to file by default

SIDs and DACLs

Security identifiers SID - identify users in Windows
rarely see it
long and complex
identify users and groups w/ SID

discretionary access control list DACL
identify who can access an

object

the DACL is a list of Access Control entries ACE - composed of a SID and permissions granted to SID

ex:
user1: Full control
user2: Read
user3: Modify

The owner establishes access
When users create files, they are designated as owner
owner can modify permissions by adding users or groups to DACL

Mandatory Access Control
mandatory access control MAC - uses labels (aka sensitivity labels or security labels) to determine access
assign labels to subjects and objects; matching labels will grant access

used in military
security-enhanced Linux SELinux
one of only OS using MAC
has 3 modes:
- enforce SELinux policy

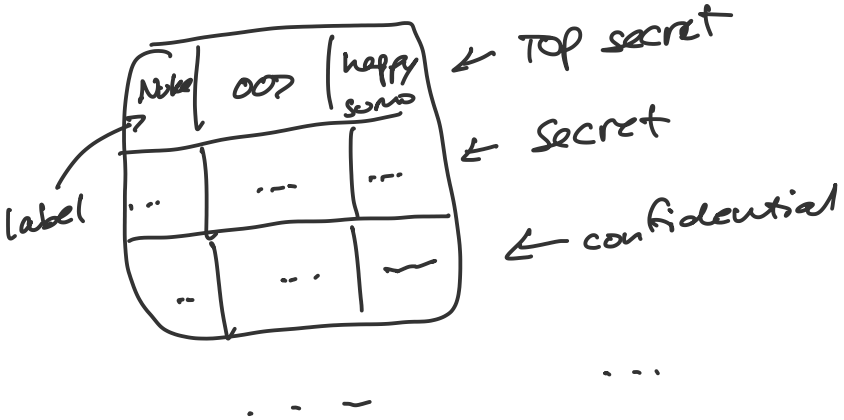
enforcing – [illegible] and ignore permissions

permissive mode – uses permissions instead of SELinux but will log any SELinux violations

disabled – does not use SELinux and does not log

Labels and Lattice

MAC uses different levels of security levels defined in lattice



label does not give access to same level labels

but can give access to lower level labels if granted access

can't be given label higher than your level

Establishing Access

admin establishes access but someone with higher authority can define access for users or objects

[illegible] access

security pro typically decides
can also upgrade/downgrade
labels are assigned through paperwork
not system
then admin assigns rights in
system

Attribute-Based Access Control ABAC

evaluates attributes and grants
access based on value of attributes

attributes can be characteristics of
user, env, resource

ex: user account may have attr
like
- employee
- inspector
- nuclear aware

a file server has a share called
'inspector' that holds docs used
by inspectors

ABAC policy would grant access
to this share if user has
all these attr

many software defined networks SDN
use ABAC

instead of rules on physical routers
policies in ABAC control traffic

ex: "Allow logged-on researchers
to access research sites on
main network"

- elements

policy statement

- subject - user; use any property as attr like employment status, group, role, logged-on status
- object - resource user trying to access
- action - action user attempting to do
- environment - everything outside subject and object attr time, location, protocols, encryption, devices
  ex. main network

ABAC is flexible and can enforce DAC and MAC

owners can create policies to grant access

uses attr that identify subjects and objects and grant access when policy identifies match

Conditional Access

Microsoft implemented conditional access into Azure Directory now

can be used with traditional schemes but added capabilities for org policies

policies - if-then statements

ex: Shares on server
docs
has other access control schemes
add on conditional access policy
that requires users to login
with MFA

Signals
- user or group membership
- IP location
- Device

Assessment Notes

1. Biometrics that collect passively, bypass enrollment, avoid obvious collection:
   - Facial
   - Gait analysis

2. Uses username/password and logs GPS location
   it doesn't mention that location used for auth"
   so this is only 2 factor

3. 2FA words that expire after 30 sec
   TOTP

4. Lowest possible CER = most accurate biometrics

5. Identify patients even if

unconscious = user alert

6. second auth factor authen app that is user friendly and non-disruptive
   push notifications

7. Time-of-day restrictions = access only during working hours

8. User account for junior admin to fix domain controller

9. remove all shared accounts to ensure logs accurately report identity of personnel in actions

10. Disable accounts that abandoned for contractors who work 1 week every quarter

11. Strong authentication and authorization for internal systems
    single username/password for ad network access.

Kerberos

OpenID = internet
SSO = web but not all

12. Web app connecting with other apps running on internet
    passive creds from an

app needs to ...
app on web domain

SAML

SSO is not always web

remote authentication dial-in user service RADIUS - AAA for remote access and wireless network solutions

13. Give 2-week notice, disable account in exit interview

14. High employee turnover, simplify account administration

group-based privileges

15. Grant users access to different shares based on job functions

role-based access control