

Access Control Models



Determine the level of access
to a resource (authorization)

Data Roles



02:10



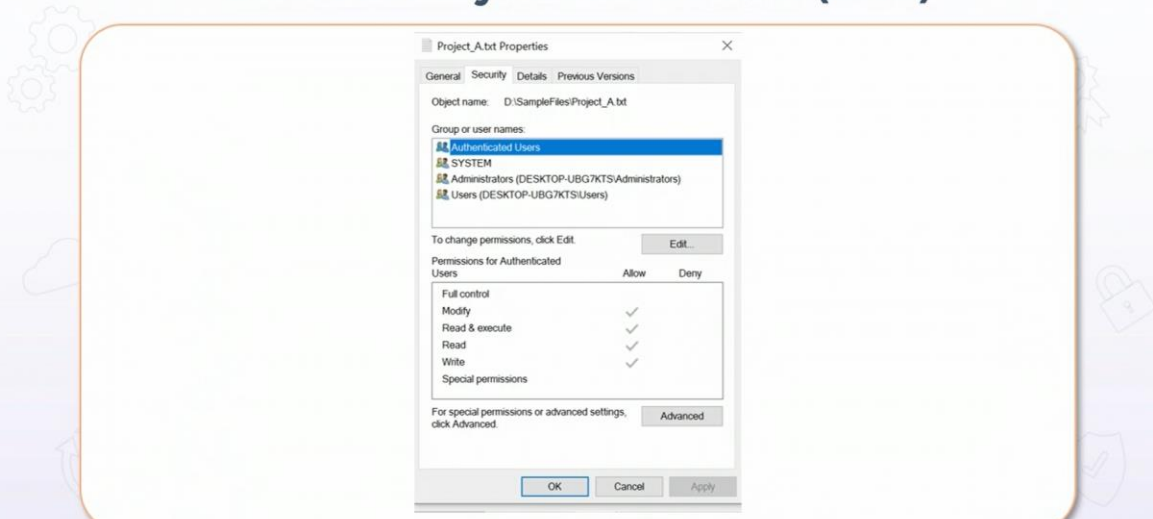
◀ Previous Topic

Next Topic ▶



06:34

Discretionary Access Control (DAC)



02:54



◀ Previous Topic

Next Topic ▶



04:50

Mandatory Access Control (MAC)



Resources (files, apps, network sockets) are assigned a label



Users are assigned clearance levels that allow access to labelled resources (controlled by OS, such as SELinux)

04:41



◀ Previous Topic

Next Topic ▶



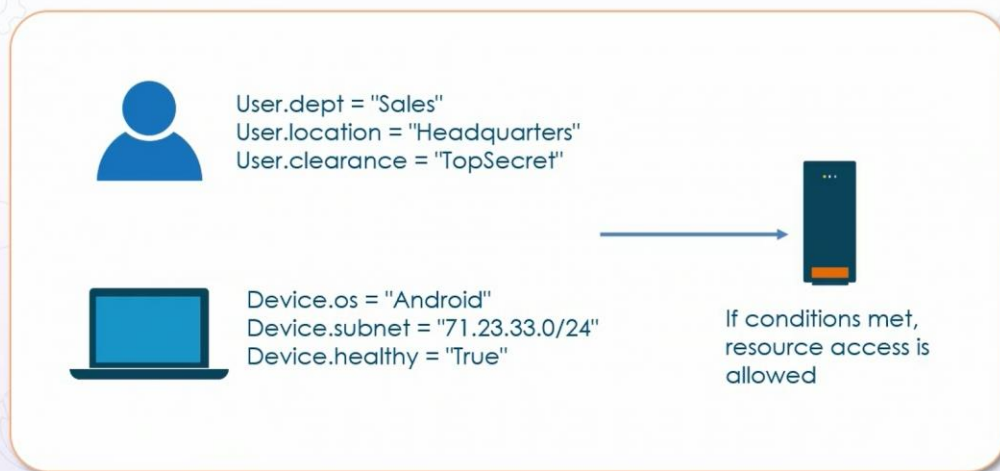
03:03

Role-based Access Control (RBAC)

The screenshot displays the Microsoft Azure portal interface for managing Role-based Access Control (RBAC) for a subscription named "Pay-As-You-Go". The left sidebar shows navigation options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Security, Events, Cost Management, Cost analysis, Cost alerts, Budgets, Advisor recommendations, Billing, and Invoices. The main content area shows the "Access control (IAM)" page with a search bar and filters. The table lists role assignments for the subscription, categorized by Contributor, Owner, and SQL DB Contributor.

Name	Type	Role	Scope
Contributor			
<input type="checkbox"/> automationaccount172_ERXa	App	Contributor	This resource
<input type="checkbox"/> User Two utwo@itdemo1outlook.com	User	Contributor	This resource
Owner			
<input type="checkbox"/> IT Demo 1 it_demo_1_outlook.com#E...	User	Owner	Management group (Inherited)
SQL DB Contributor			
<input type="checkbox"/> Identity not found. Unable to find identity.	Unknown	SQL DB Contributor	This resource

Attribute-based Access Control (ABAC)



07:29



Previous Topic

Next Topic



00:15

Broken Access Control Attacks



Allow unauthorized resource access

Broken Access Control Attacks

Access Control



Occurs after successful authentication

Ensures user access to resources

Adheres to the principle of least privilege

01:47




Previous Topic

Next Topic

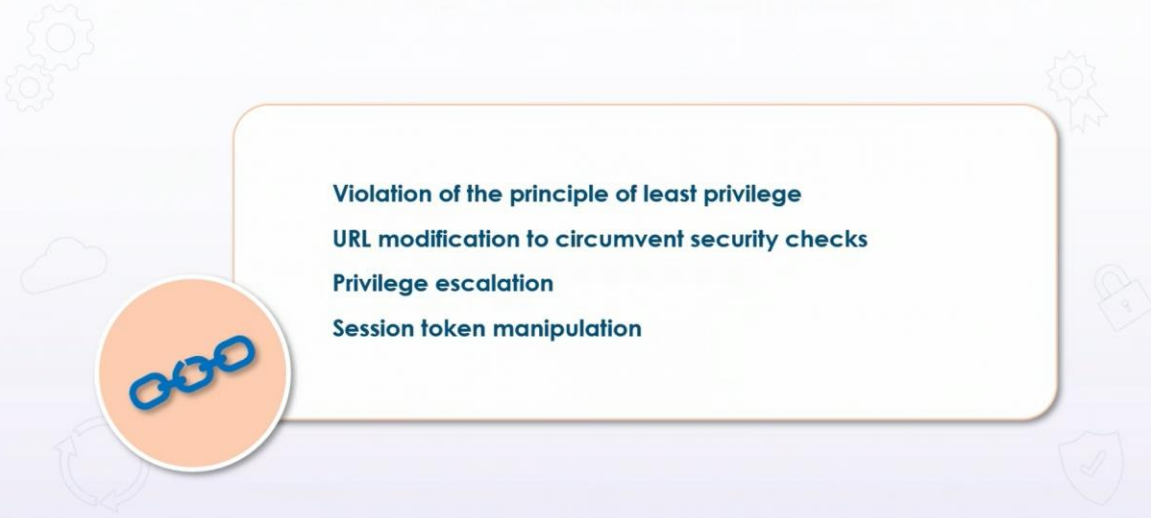
04:59



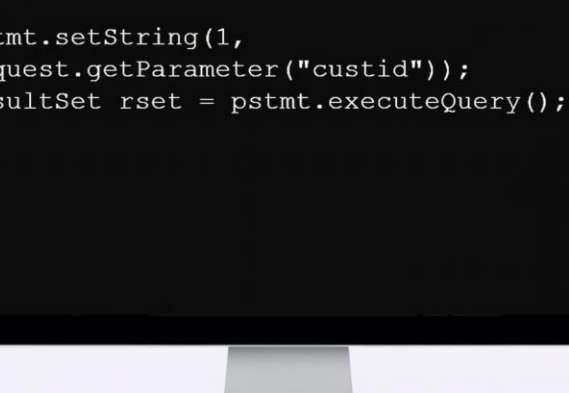
Broken Access Control Attacks



- Violation of the principle of least privilege
- URL modification to circumvent security checks
- Privilege escalation
- Session token manipulation



Unverified User Input

A computer monitor with a black bezel and a silver stand. The screen is black and displays a Java code snippet in a white, monospaced font. The code is as follows:

```
pstmt.setString(1,  
request.getParameter("custid"));  
ResultSet rset = pstmt.executeQuery();
```



Forced URL Browsing



Unauthenticated access to web pages

Authenticated regular user access to admin web pages

Client-side Input Validation



Potentially controlled by user
before submitting to server

HTTP Methods



Request methods

Define action to be run by an HTTP web
server

HTTP HEAD Method



Ask for HTTP headers as if an actual HTTP GET method was executed

E.g. request content-length before downloading a potentially large file

HTTP GET Method



Get data from HTTP server



HTTP_FTP_PacketCapture.pcap

No.	Time	Source	Destination	Protocol	Length	Info
399	24.951448	192.168.1.157	192.168.1.162	HTTP	386	GET /favicon.ico HTTP/1.1
400	24.956750	192.168.1.162	192.168.1.157	HTTP	1436	HTTP/1.1 404 Not Found (text/html)

HTTP POST Method



Send HTML form data to HTTP server



http_clear_authn_traffic_packet45.pcapng

No.	Time	Source	Destination	Protocol	Length	Info
45	22.956514659	192.168.4.250	192.168.4.24	HTTP	116	HTTP/1.1 304 Not Modified
53	23.1315601957	192.168.4.24	192.168.4.250	HTTP	685	POST /ui/panel/ hooked-browser-tree-update.json HTTP/1.1
55	23.139931298	192.168.4.250	192.168.4.24	HTTP/35...	270	HTTP/35...
60	31.187940389	192.168.4.24	192.168.4.250	HTTP	685	POST /ui/panel/ hooked-browser-tree-update.json HTTP/1.1

HTTP PUT Method



Replace server resource (new or update)

E.g. PUT /quick24x7_forums/newcomment
HTTP/2.0

Other HTTP Methods

- **DELETE**
- **CONNECT**
 - Establish tunnel to HTTP server
- **OPTIONS**
 - Show info regarding communication options
- **TRACE**
 - Loop-back testing
- **PATCH**
 - Modify resource

Mitigating Broken Access Control Attacks



Periodic authentication and permissions review

Web Application Firewall (WAF)



Designed to look for web application attacks



Can prevent and report on potential web application access control activity

Broken Access Control Mitigation

Deny by default



Least privilege



Expire tokens upon session tear-down



Limit passing data via URL



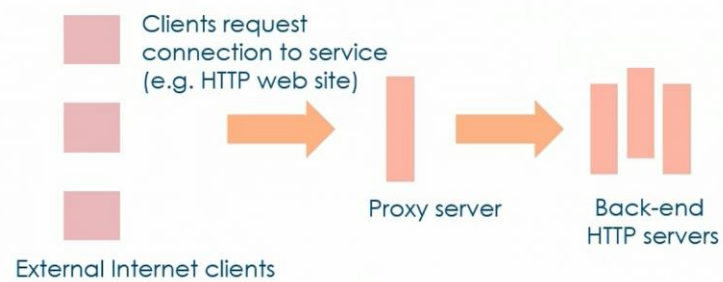
Broken Access Control Mitigation



Do not expose hosts directly to the Internet

Use dedicated management VLANs

Reverse Proxying



Insecure Wi-Fi Implementation

```
Aircrack-ng 1.6

[00:00:00] 4/4 keys tested (125.37 k/s)

Time left: --

KEY FOUND! [ 2W4335102288 ]

Master Key      : 5E D7 7D A5 FD A3 01 4E FB BB 83 F6 51 65 28 BE
                  B7 24 02 05 7A 53 9A 02 E4 09 B6 39 7B 05 09 5D

Transient Key   : CE F6 18 3E E9 51 25 BA 3C 96 44 37 F3 99 CF F0
                  E1 E1 B7 52 E3 00 00 EB 45 23 6F FE EE BE F4 66
                  BF 1E 2D 87 F3 9E 31 4D F4 1C 8B B3 91 39 F8 ED
                  0B CC B4 29 1C 8C B0 94 D2 4D CE A3 93 00 00 00

EAPOL HMAC     : 51 9D 11 C9 6C 4B B0 D5 0E DE CE 25 48 88 A0 69
```