

u2u U2U Online

online.u2u.be/courses/25896/modules/13/24863

Home > Web Security Fundamentals Techniques > Security Misconfiguration

u2u

Developer and IT Training

Security Misconfiguration

OWASP Web Security Threat #5

25°C Mostly cloudy

Search

1

The screenshot shows a web browser window titled "U2U Online" with the URL "online.u2u.be/courses/25896/modules/13/24863". The page content is a slide titled "Agenda" with a red horizontal line below it. To the left is a sidebar with a navigation tree under "Security Misconfiguration". The main content area contains a bulleted agenda. A large "u2u" logo is in the top right corner. The bottom of the screen shows a Windows taskbar with various icons and system status.

Agenda

- Introduction
- Turning off unnecessary features
- Running with **Least Privilege**
- Running SQL server with Least Privilege
- Using **Impersonation**
- Enable **CORS** correctly

#5: Security Misconfiguration

- For Good Security you need a Secure Configuration
- A Secure Configuration is necessary
 - For the application
 - For the frameworks
 - For the application server, web server and so on
- How do you define a secure configuration?
 - Keep in mind the most optimal settings for security
 - Keep software up to date
 - Steer away from default settings (e.g., default password for a platform)

u2u U2U Online

online.u2u.be/courses/25896/modules/13/24863

Home > Web Security Fundamentals Techniques > Security Misconfiguration

Security Misconfiguration

- Introduction
- Turning off unnecessary features
- Setting up incorrect storage
- Running SSL/TLS server with least privilege
- Using implementation
- CORES
- Summary

Security Matrix

u2u

ATTACK VECTORS	SECURITY WEAKNESS	TECHNICAL IMPACTS
Exploitability	Prevalence	Detectability
EASY	WIDESPREAD	EASY
		MODERATE

25°C Mostly cloudy

Search

6:41 PM 12/29/2023

u2u U2U Online

online.u2u.be/courses/25896/modules/13/24863

Home > Web Security Fundamentals Techniques > Security Misconfiguration

Security Misconfiguration

- Introduction
- Turning off unnecessary features
- Setting up correct storage
- Running SSL/TLS with least privilege
- Using implementation
- CORS
- Summary

Visualizing an Attack

u2u

The diagram shows a stick figure on the left and a server rack icon on the right. A red arrow points from the stick figure to the server rack, labeled "Tries to find a vulnerability". Another red arrow points from the server rack back to the stick figure, labeled "Responds with a possible opening".

25°C Mostly cloudy

Search

6:41 PM 12/29/2023

u2u Online

online.u2u.be/courses/25896/modules/13/24863

Defenses

- Decrease the attack surface – everything off by default
 - Turn off unnecessary features
 - In the OS, database
 - And in your software! Is this feature essential?
 - Use accounts with least permission
- Optimize Application Security Config
 - Don't use debug configuration
- Keep Software (OS, database, libraries, etc...) up to date
- Be careful about defaults, such as username/passwords
 - There are a lot of devices out there with their defaults still set

25°C Mostly cloudy 6:41 PM 12/29/2023

u2u Online

online.u2u.be/courses/25896/modules/13/24863

Turning off unnecessary features



- Decrease the attack surface of your web site
- This will most likely also speed up your web site
- Remove unneeded ASP.NET modules
- Disable unused view engines
- Disable debugging and run in release

ASP.NET Core takes care of most of these issues for us

25°C Mostly cloudy

Search

6:41 PM 12/29/2023

Remove ASP and MVC version headers

- Running ASP.NET Core on Kestrel exposes **Server: Kestrel**

```
builder.WebHost.ConfigureKestrel(options=>options.AddServerHeader = false);
```
- Running ASP.NET Core on IIS still exposes the **X-Powered-By** header
 - Remove in web.config

```
<system.webServer>
  <httpProtocol>
    <customHeaders>
      <remove name="X-Powered-By" />
    </customHeaders>
  </httpProtocol>
</system.webServer>
```

The screenshot shows a web browser window with the URL online.u2u.be/courses/25896/modules/13/24863. The page title is "Running with Least Privilege". The left sidebar contains a navigation menu with items like "Home", "U2U Online", "My profile", "Your review", "Send an review", "Security Misconfiguration", "Security Misconfiguration", "Introduction", "Turning off unnecessary features", "Setting up least privilege", "Running SQL Server with Least Privilege", "Using impersonation", "COMS", and "Summary". The main content area has a large heading "Running with Least Privilege" and a bulleted list under it. The bottom of the screen shows a taskbar with various icons and system status information.

Running with Least Privilege

- Never give your application more privileges than needed to do the job!
 - This way security flaws are less exploitable
 - For example, updating a table with SQL injection won't work
 - A Trojan horse or virus cannot write to Program Files if not running as local admin
 - A hacker cannot deface your web site if they cannot write to the website's folder

25°C Mostly cloudy 6:41 PM 12/29/2023

The screenshot shows a web browser window with the URL online.u2u.be/courses/25896/modules/13/24863. The page title is "Discretionary Access Control". The left sidebar displays a navigation tree under "Security Misconfiguration" with nodes like "Introduction", "Turning off unnecessary features", "Setting file and folder privileges", "Using impersonation", "Core", and "Summary". The right sidebar features the "u2u" logo. The bottom of the screen shows a Windows taskbar with icons for Start, Search, File Explorer, Task View, Edge, File Explorer, Google Chrome, and a battery icon. The system tray shows the date and time as 12/29/2023 at 6:41 PM.

Discretionary Access Control

- The Windows OS protects securable resources with DACLs
 - Discretionary Access Control Lists
 - Which are a series of Access Control Entries (ACEs)
- When a user logs into windows a token is created for that user
 - That token is applied to every process (and thread) created by that user
 - The token contains SIDs and privileges
 - SIDs are used to perform access checks against ACLs on resources
 - Privileges are used to perform specific machine wide tasks, such as backup

A Process for Determining Appropriate Privilege

- Find out each resource the application uses, with kind of access
 - Files, registry, active directory data, etc.
- Find out each privileged API the application uses
 - Create file in upload directory, create process
- Determine which account the application will use
 - Don't be lazy and immediately assume using a local admin account
- Configure the account
 - This is admin's work
- Configure your webserver to use this account
 - Configure an **ApplicationPool** in IIS to use this account
 - Or use **impersonation**

The screenshot shows a web browser window with the URL online.u2u.be/courses/25896/modules/13/24863. The page title is "Running SQL Server with Least Privilege". The left sidebar shows a navigation tree under "Security Misconfiguration" with nodes like "Introduction", "Turning off unnecessary features", "Setting SQL Server with Least Privilege", "Using impersonation", "COMs", and "Summary". The right sidebar has a large "u2u" logo. The bottom of the screen shows a Windows taskbar with icons for Start, Search, File Explorer, Task View, Edge, Google Chrome, and other system icons. The system tray shows the date and time as 6:41 PM on 12/29/2023.

Running SQL Server with Least Privilege

- Use **Windows Integrated Security** for logging into the database
 - Create an account for the application
 - No need to store any password in the connection string
- If you need to use SQL Server authentication
 - Never run with the **sa** account (never ever!)
- Define Login, User and Role
 - And assign permissions only for what is needed
- Disable any unused features

The screenshot shows a web browser window with the URL online.u2u.be/courses/25896/modules/13/24863. The page title is "Creating an SQL Server Application Login". The left sidebar shows a navigation tree under "Security Misconfiguration" with sections like "Introduction", "Turning off unnecessary features", "Running SQL Server with Least Privileges", "Using impersonation", "COM", and "Summary". The right sidebar has a large "u2u" logo. The bottom of the screen shows a taskbar with icons for weather (25°C, Mostly cloudy), search, file explorer, and browser.

Creating an SQL Server Application Login

- You need a login to connect to SQL Server
- You can create logins based on
 - Windows Integrated Principal (with Active Directory)
 - SQL Server
- Logins work at the level of the server
 - Not the database level, for this you need a user
 - Can have a default database

Creating a SQL Server login

- Use SQL Server Management Studio
- Or script

```
USE [master]
GO
CREATE LOGIN [Joske]
    FROM WINDOWS
    WITH DEFAULT_DATABASE=[master]
GO
```

```
USE [master]
GO
CREATE LOGIN [Jefke]
    WITH PASSWORD=N'secret',
    DEFAULT_DATABASE=[master],
    CHECK_EXPIRATION=OFF,
    CHECK_POLICY=OFF
GO
```

Server->Security->Logins

Login name: Search...

Windows authentication
 SQL Server authentication

Password:
 Confirm password:

Specify old password
 Old password:

Enforce password policy
 Enforce password expiration
 User must change password at next login

Mapped to certificate
 Mapped to asymmetric key
 Map to Credential

Mapped Credentials:

Credential	Provider

Add Remove

Default database: master

Default language: <default>

Creating a SQL Server User

- Logins must be mapped to a user to access a database
- One login can be mapped to different databases with different users
 - But can only map to one user in a specific database
- Permissions are granted (or denied) to users in a database

25°C Mostly cloudy Search 6:41 PM

Creating a SQL Server User

- Using SSMS
- Or using script

```
USE [Northwind]
GO
CREATE USER [Joske] FOR LOGIN [Joske]
GO
```

User type:
SQL user with login

User name:
Joske

Login name:
Joske

Default schema:

- Then you still need to assign a role

The screenshot shows a web browser window with the URL online.u2u.be/courses/25896/modules/13/24863. The page title is "Creating a SQL Server Role". The left sidebar contains a navigation tree under "Security Misconfiguration" with nodes like "Introduction", "Turning off unnecessary features", "Setting up a user account", "Using impersonation", "CORES", and "Summary". The right sidebar features a large "u2u" logo. The bottom of the screen shows a Windows taskbar with icons for Start, Search, File Explorer, Task View, Edge, Google Chrome, and File Explorer. The system tray shows the date and time as 12/29/2023 at 6:41 PM.

Creating a SQL Server Role

- SQL Server roles are like Active Directory Groups
- You use SQL roles to efficiently manage permissions
 - Assign permissions to roles
 - Then assign users to roles
- Two kinds of custom roles
 - Database roles
 - Gives same permissions to user as to application, user can still use management studio
 - Application roles
 - Only gives permissions to the application, not the user

Creating a SQL Server Role

- Using SSMS
 - Select role name
 - Add owned schemas
 - Add members
- Or using script

```
USE [Northwind]
GO
CREATE ROLE [MyApp]
GO
USE [Northwind]
GO
ALTER ROLE [MyApp] ADD MEMBER [Jefke]
GO
```

The screenshot shows a browser window for 'u2u Online' with the URL 'online.u2u.be/courses/25896/modules/13/24863'. The main content area displays a slide titled 'Creating a SQL Server Role'. On the left, there's a sidebar with navigation links like 'Home', 'About', 'Contact', 'Logout', and 'My courses'. The main content area has a large title 'Creating a SQL Server Role' and two bullet points: 'Using SSMS' and 'Or using script'. Under 'Using SSMS', there's a list of three steps: 'Select role name', 'Add owned schemas', and 'Add members'. Below this is a block of T-SQL script:

```
USE [Northwind]
GO
CREATE ROLE [MyApp]
GO
USE [Northwind]
GO
ALTER ROLE [MyApp] ADD MEMBER [Jefke]
GO
```

On the right, there's a screenshot of the 'Role Properties' dialog box for the 'MyApp' role. The 'Role name:' field is set to 'MyApp'. The 'Owner:' field is empty. The 'Schemas owned by this role:' section lists several schemas: db_accessadmin, dbo, db_securityadmin, sys, db_owner, and db_backupoperator. The 'Members of this role:' section shows one member named 'Jefke'. At the bottom of the dialog are 'Add...' and 'Remove' buttons.

Adding Securables to a Role

- **Securables** are the resources SQL server authorization regulates
 - For example, a table is a securable
- You assign **permissions** to users for a certain securable
 - GRANT
 - DENY

25°C Mostly cloudy 6:41 PM ENG 12/29/2023

Adding Securables to a Role

- Using SSMS
 - Select a securable with Search
 - Then configure permissions
- Using script

```
use [Northwind]
GO
GRANT SELECT ON [dbo].[Categories]
  TO [MyApp]
GO
use [Northwind]
GO
DENY ALTER ON [dbo].[Categories]
  TO [MyApp]
GO
```

Permission	Grantor	Grant	With Grant	Deny
Alter		<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Control		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Delete		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Insert		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
References		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Select		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
T-SQL permissions		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

The screenshot shows a web browser window for 'u2u Online' with the URL online.u2u.be/courses/25896/modules/13/24863. The page title is 'Security Tip'. The main content area contains a list of security tips:

- Never give developers access to a table
 - Instead use a view
- This way you can update the view to filter which data the application can see

The browser interface includes a sidebar with user information (Maged Alkhatib, MagedAlkhatib), a navigation menu (Home, Security Misconfiguration, Security Misconfiguration Techniques, Security Misconfiguration Examples, Introduction, Turning off unnecessary features, Setting database privileges, Using impersonation, CORS, Summary), and a toolbar with various icons. The system tray at the bottom shows the date and time (6:41 PM, 12/29/2023) and battery status.

U2U Online

online.u2u.be/courses/25896/modules/13/24863

Identity flow for internet web sites

The diagram illustrates the identity flow for internet web sites. It shows two users, a woman and a man, performing a search on their web browsers for "trovi". Their search requests are sent to a central server. The server then queries a database for user information.

u2u U2U Online

online.u2u.be/courses/25896/modules/13/24863

Home > Web Security Best Practice Techniques > Security Misconfiguration

Security Misconfiguration

- ↳ Mapped Drives
- ↳ Microsoft IIS
- ↳ Inheritance
- ↳ Turning off unnecessary features
- ↳ Setting static content storage
- ↳ Running IIS on port 80
- ↳ Impersonation
- ↳ CORS
- ↳ Summary

Intranet: Using Impersonation

The diagram illustrates a network architecture involving three main components: a client browser, a central server, and a database. On the left, two user icons (a woman and a man) are shown performing a search on a 'trolli' website. Arrows from both users point to a central server icon, which is depicted as a grey rectangular box with three horizontal slots and a blue circular base. From the central server, arrows point to both the woman and the man, and finally to a large blue cylinder representing a database. A small orange gear icon is positioned near the base of the central server, indicating the mechanism for impersonation.

25°C Mostly cloudy

6:41 PM 12/29/2023

ENG

The screenshot shows a web browser window with the URL online.u2u.be/courses/25896/modules/13/24863. The page title is "Enabling Impersonation". On the left, there's a sidebar with a navigation tree under "Security Misconfiguration". The main content area contains two bullet points and some code snippets.

- For the whole site

```
<authentication mode="Windows" />
<identity impersonate="true" />
```

- For specific operations

```
WindowsPrincipal wp = User as WindowsPrincipal;
WindowsIdentity wi = wp.Identity as WindowsIdentity;

using (var impersonationContext = wi.Impersonate()) {
    ...
}
```

At the bottom, the taskbar shows the weather (25°C, Mostly cloudy), search bar, pinned icons for File Explorer, Task View, File History, Task Scheduler, Task Manager, and Google Chrome, and system status icons for battery, signal, and volume.

Same Origin Policy (SOP)

- Critical Security Mechanism restricting how a document loaded from one origin can interact with resource from another origin
- Isolates potential malicious documents
- Ex.: Trying to execute JavaScript to load a JavaScript file from another domain
 - JS on <https://www.u2u.be> tries to download/execute JS from <https://www.microsoft.com>

The screenshot shows a Microsoft Edge browser window with the URL online.u2u.be/courses/25896/modules/13/24863. The main content area displays a slide titled "Same Origin Policy" with a red header bar. Below the title is a bulleted list:

- SOP checks current URL with the other URL and then decides the outcome
- In the following example we check a URL with another one from the table:
<http://store.company.com/dir/page.html>

Below the list is a table comparing URLs and their outcomes:

URL	Outcome	Reason
http://store.company.com/dir2/other.html	Same origin	Only the path differs
http://store.company.com/dir/inner/another.html	Same origin	Only the path differs
https://store.company.com/page.html	Failure	Different protocol
http://store.company.com:81/dir/page.html	Failure	Different port (http:// is port 80 by default)
http://news.company.com/dir/page.html	Failure	Different host

The browser's taskbar at the bottom shows various pinned icons and the system tray with the date and time.

The screenshot shows a web browser window with the URL online.u2u.be/courses/25896/modules/13/24863. The page title is "CORS". The left sidebar lists "Security Misconfiguration" topics: "Mixed Content", "Security Misconfiguration", "Innovation", "Turning off unnecessary features", "Setting strict cookie flags", "Turning SSL/TLS server-side Load Balancer", "Using implementation", and "CORS". The right sidebar features the "u2u" logo. The main content area contains a bulleted list:

- Cross-Origin Resource Sharing
- By adding specific headers, the SOP can be made more permissive
 - Access-Control-Allow-Origin
 - Access-Control-Allow-Credentials
 - Access-Control-Allow-Headers
 - Access-Control-Allow-Methods
 - ...
- The server needs to add these headers to the response message

The taskbar at the bottom shows the Windows Start button, a search bar, pinned icons for File Explorer, Task View, Control Panel, Internet Explorer, Google Chrome, and Microsoft Edge, and system status icons for battery, signal, volume, and date/time.

Access-Control-Allow-Origin

- Can the response be shared with requesting code from the given origin
- Syntax
 - Access-Control-Allow-Origin: *
 - Access-Control-Allow-Origin: <origin>
 - Access-Control-Allow-Origin: null
- The literal value “*” is the only wildcard available, allowing any origin
 - Cannot be used when Credentials are Allowed
- Origin only allows one single value, not a list
 - Make sure to check the Origin request header, validate it and then set it for the response

25°C Mostly cloudy 6:41 PM

u2u U2U Online

online.u2u.be/courses/25896/modules/13/24863

Access-Control-Allow-Credentials

- Expose the response to JS when request's credentials mode is “include”
 - Credentials are cookies, authorization headers or TLS client certificates
- Syntax (omit it when not necessary)
Access-Control-Allow-Credentials: true

25°C Mostly cloudy

Search

6:41 PM 12/29/2023

What can go wrong?

- ACAO only allows one domain, but what if we want more?
 - Use the * wildcard (but we might need Credentials to work...)
 - **Use the domain the user is giving us**
- HTTP GET to btc-exchange.com

```
GET /api/requestApiKey HTTP/1.1
Host: btc-exchange.com
Origin: http://Labs-albinowax
Cookie: sessionid=validSessionId
```

- Response from btc-exchange.com

```
HTTP/1.1 200 OK
Access-Control-Allow-Origin: http://Labs-albinowax
Access-Control-Allow-Credentials: true

{"id":"zv691C...
```

How do we exploit this?

The diagram illustrates a security exploit. A user is shown at a computer, with a briefcase containing a skull and crossbones icon nearby. A flower is on the desk. The user is interacting with a laptop. To the right, a database labeled "Admin" contains several records. A red box labeled "? Sensitive data" points to one of these records. Below the laptop, a red box labeled "API key" points to the same record. Above the laptop, a request is shown:

```
GET /sensitive-victim-data HTTP/1.1  
Host: vulnerable-website.com  
Origin: https://malicious-website.com  
Cookie: sessionid=...
```

Below the response is:

```
HTTP/1.1 200 OK  
Access-Control-Allow-Origin: https://malicious-website.com  
Access-Control-Allow-Credentials: true
```

At the bottom of the slide, there is a navigation bar with icons for search, file, and other applications, along with system status information including weather (25°C, Mostly cloudy), time (6:41 PM), and date (12/29/2023).

The screenshot shows a web browser window with the URL online.u2u.be/courses/25896/modules/13/24863. The page title is "Generating Dynamic ACAO". The left sidebar shows a navigation tree under "Security Misconfiguration" with nodes like "Security Misconfiguration", "Innovation", "Turning off unnecessary features", "Setting up account storage", "Running SSL/TLS with Let's Encrypt", "Using implementation", and "Summary". The right sidebar has a "u2u" logo. The bottom taskbar shows the Windows Start button, a search bar, pinned icons for File Explorer, Task View, Control Panel, Internet Explorer, Google Chrome, and Microsoft Edge, and system status icons for weather (25°C, mostly cloudy), battery, network, volume, and date/time (6:41 PM, 12/29/2023).

Generating Dynamic ACAO

- If you need to create a dynamic ACAO header
 - Make sure to validate the Origin header coming from the browser
- Validation?
 - Best way is to register and validate every domain/subdomain in full!
- Half measure
 - Only check if the domain ends with google.com (evil.google.com is fine – can be abused from XSS vulnerable legacy applications)
 - Check if the domain starts with <https://btc.net> (<https://btc.net.evil.net> is fine)
 - Not checking the protocol (HTTP connections are allowed – can be abused with MiTM)

Best Practices

- “Access-Control-Allow-Origin: null” is a valid value
 - Works in the same way as the wildcard
 - Is even more dangerous (ACAC is available!)
- Use “Vary: Origin” response header
 - To avoid caching issues
 - To make sure an exploit called “cache poisoning” is not feasible

25°C Mostly cloudy 6:41 PM 12/29/2023

Summary

- Turn off unneeded features on your machines
 - Unused modules, view engines
 - Disable debugging features, such as logging
 - Turn on retail mode
- Run your processes with least privileges
 - Configure an account for your application
 - Configure files and folders with the proper ACLs for that account
 - Configure the SQL server account to also use least privilege
- Pros and cons of using Impersonation
- SOP & CORS

u2u U2U Online

online.u2u.be/courses/25896/modules/13/24863

Home > Web Security Best Practice Techniques > Security Misconfiguration

My profile

Logout

My courses

Developer and IT Training

Security Misconfiguration

- Introduction
- Taking off unnecessary privileges
- Setting up least privilege
- Running SQL Server with Least Privilege
- Using impersonation
- CORS
- Summary

u2u

Lab

Running with Least Privilege

25°C Mostly cloudy

Search

6:41 PM 12/29/2023