# Nizar Session

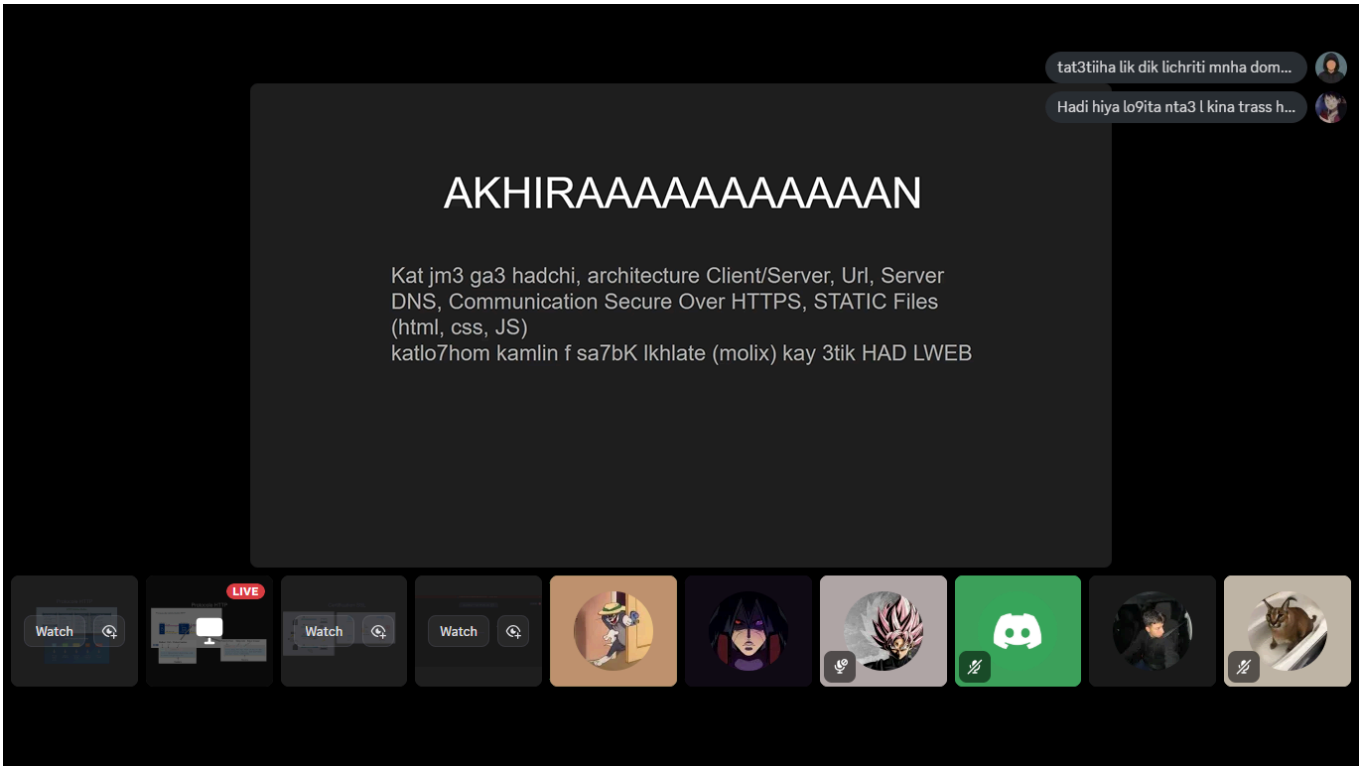
# DNS

- In a URL, we have TLD (Domaine de premier niveau) (Top Level domain), it is important. (It is usually the final part of the url (.org, .ma, .com, ...))
- You can see all DNS with this website : https://securitytrails.com/

# HTTPS

- Fun Fact, HTTPS = HTTP + SSL Encryption.
- SSL is used with a certification, SSL certification.

# In general

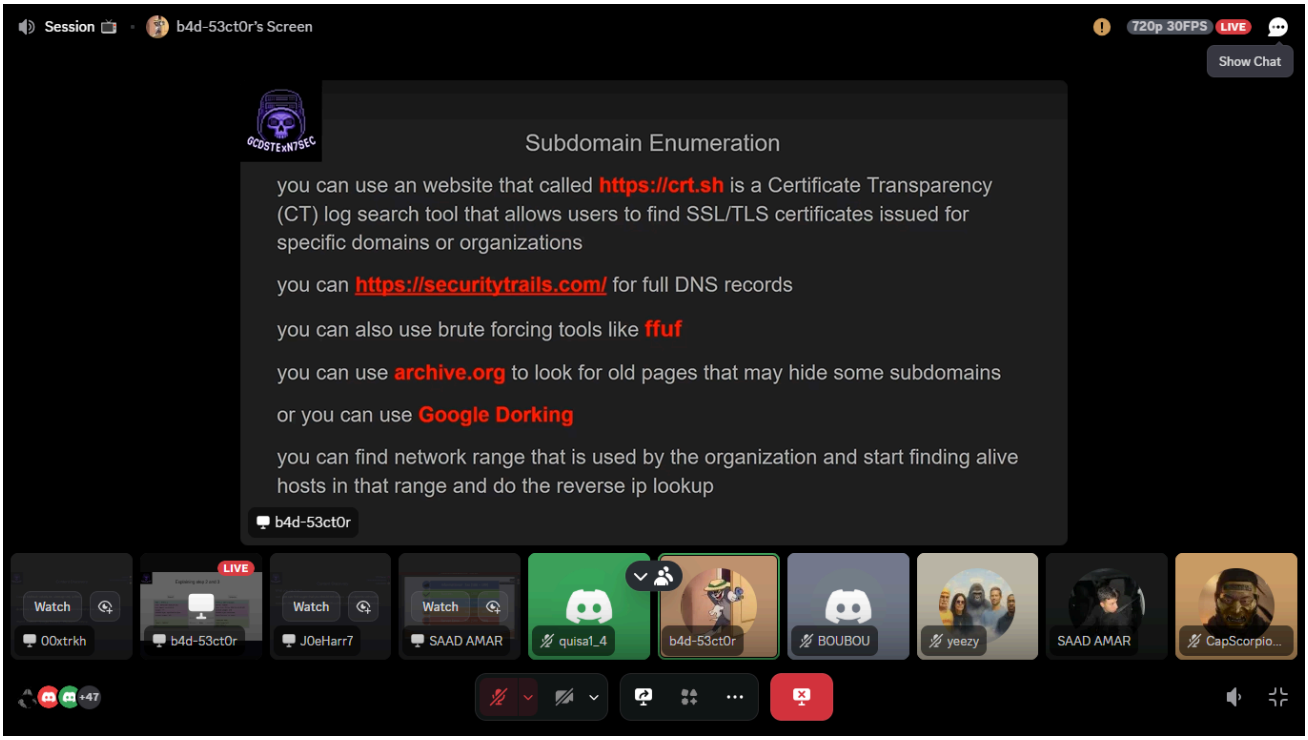



# OSINT

## - Google dorking,

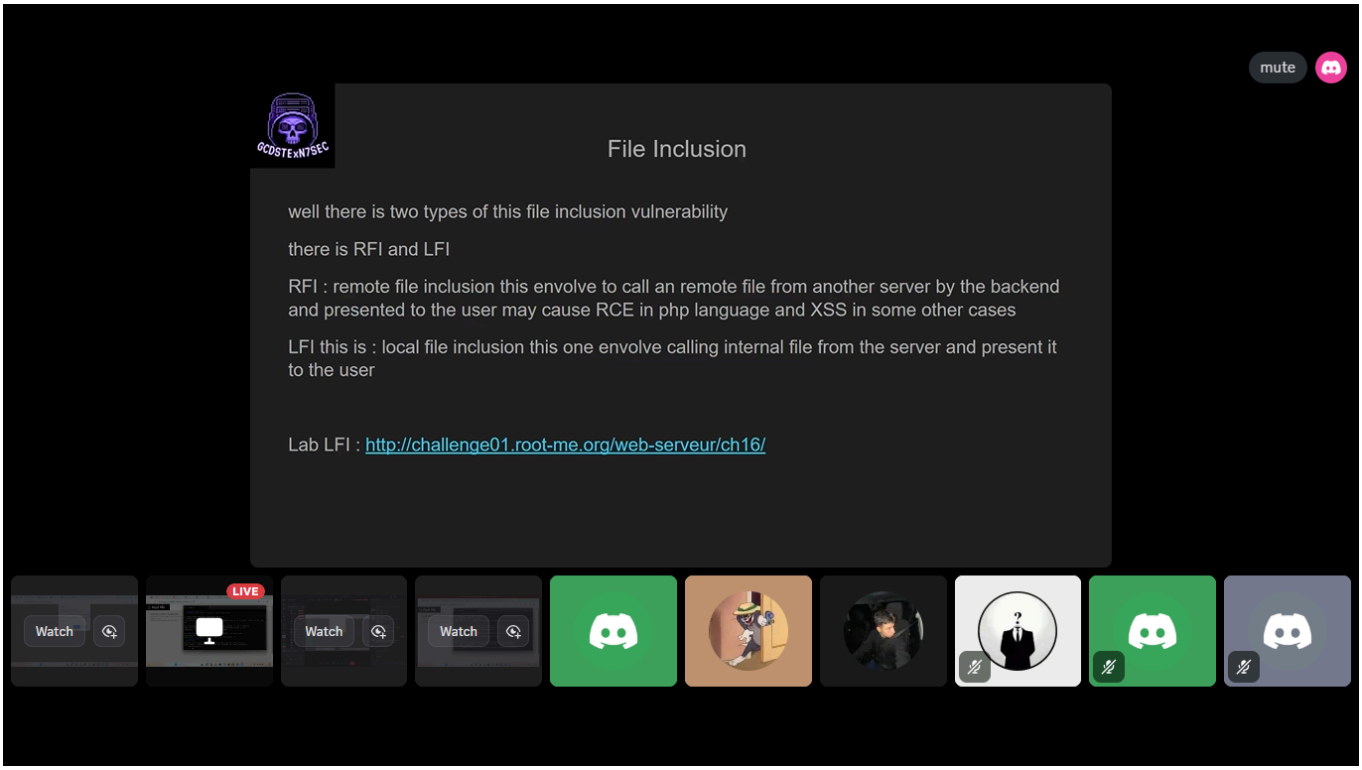
also known as Google hacking, is a technique that uses advanced Google Search queries to find security vulnerabilities in websites. It involves using specific keywords and operators to

locate exposed information or misconfigured systems, such as finding sensitive files or login pages that were not intended to be public.




- It's here where we use the Authentification ByPass

## - File Inclusion:




RFI and LFI are both **file inclusion vulnerabilities** in web applications that occur due to improper input validation, but they differ in the **location of the files** an attacker can include.

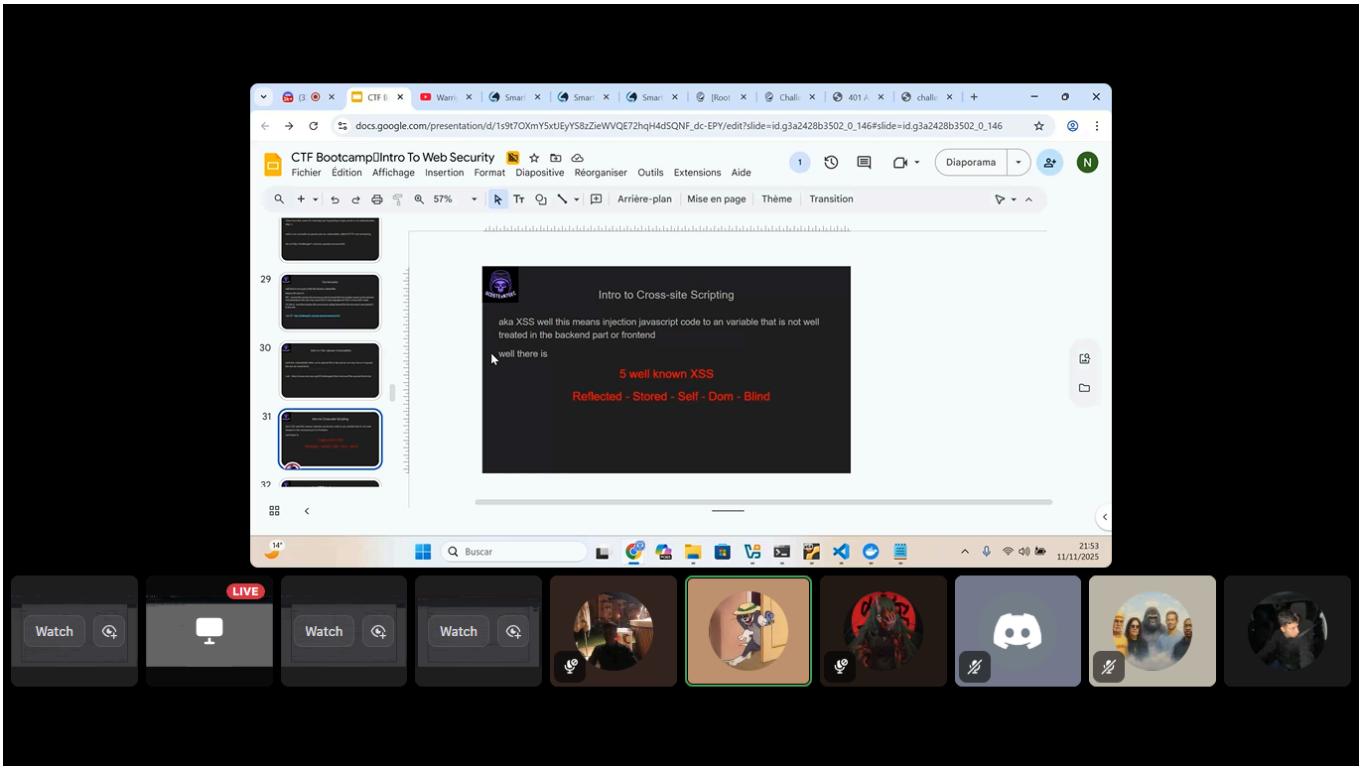
RFI: Remote File Inclusion

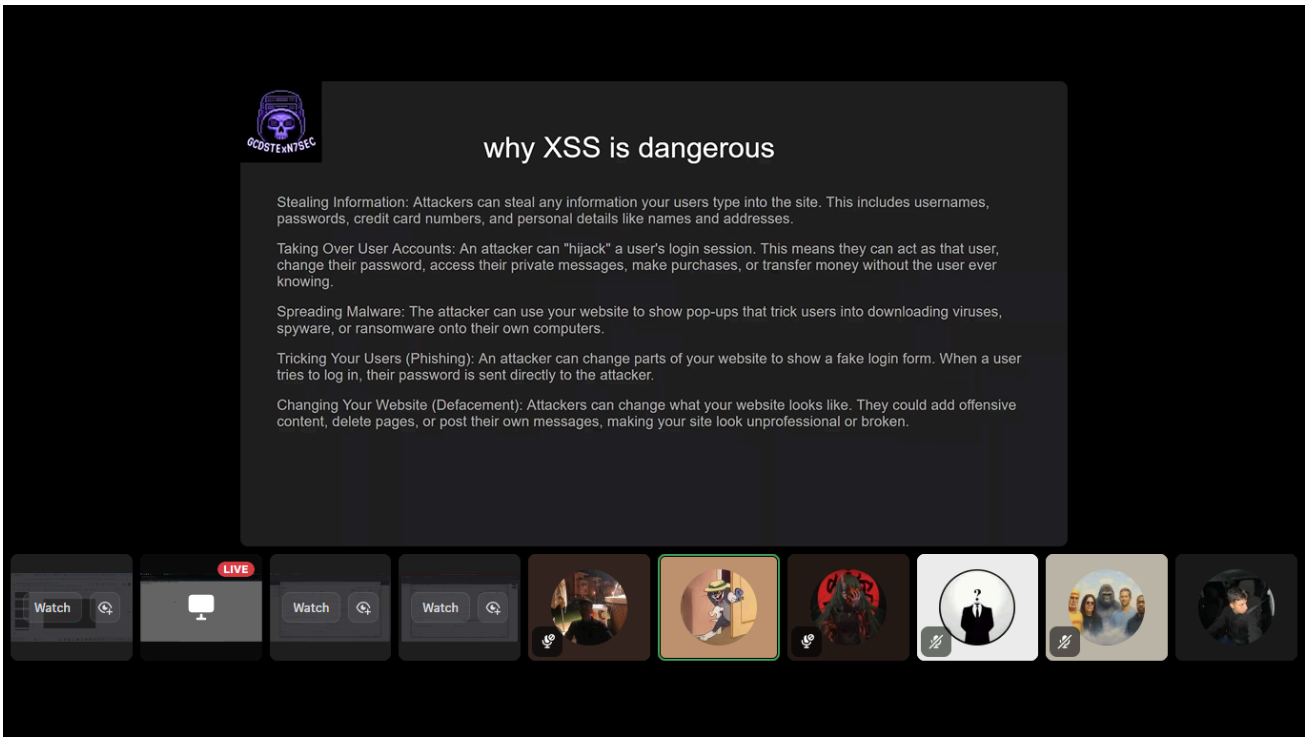
- **Definition**: RFI allows an attacker to include files hosted on a **remote server** (an external source controlled by the attacker) into the vulnerable web application.
- **Mechanism**: The web application fetches and executes the attacker's remote malicious code, typically specified via a URL in a user-supplied input parameter.
- **Severity**: RFI is generally considered more severe than LFI because it can immediately lead to **remote code execution (RCE)** on the target server, allowing the attacker to run arbitrary commands and potentially compromise the entire system.
- **Prerequisites**: A successful RFI attack often requires specific server configurations, such as the `allow_url_include` setting being enabled in PHP.

LFI: Local File Inclusion

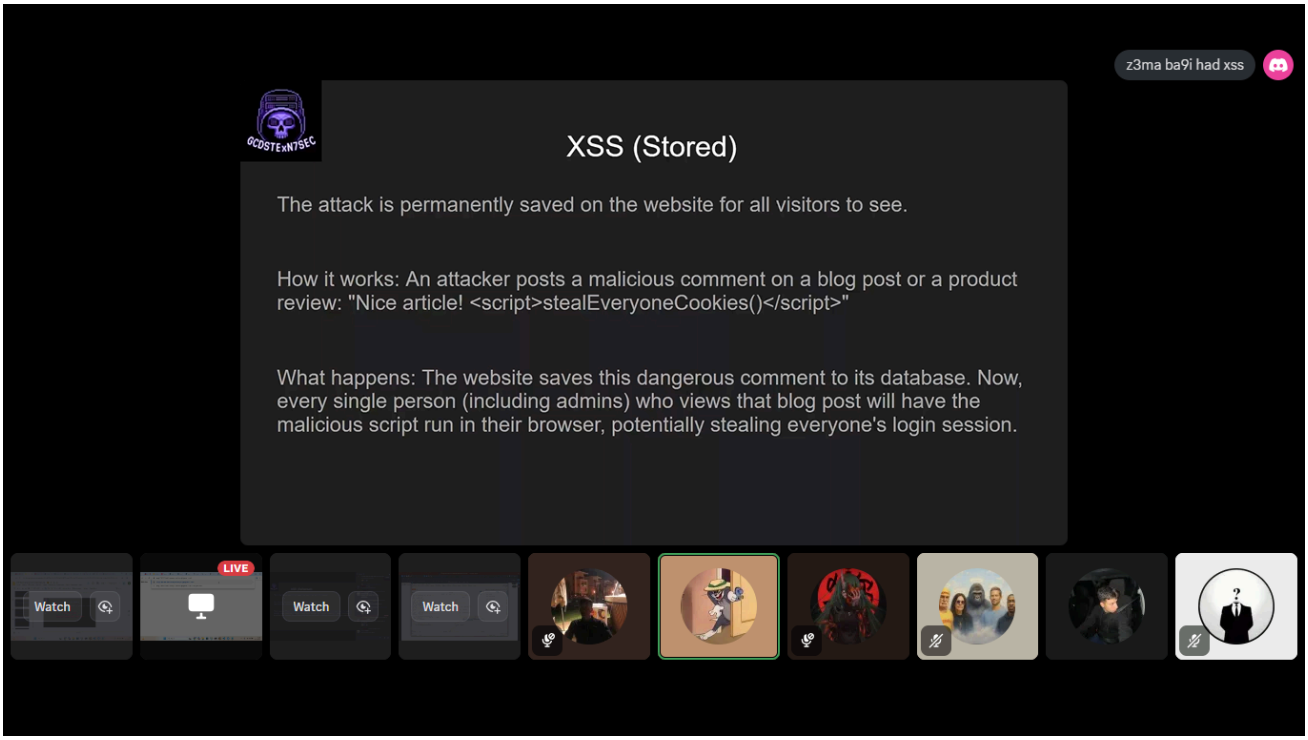
- **Definition**: LFI allows an attacker to include and access files that are already present on the **local server** where the application is running.
- **Mechanism**: Attackers manipulate file paths using techniques like directory traversal (e.g., `../../`) in user input to access sensitive local files (like configuration files, source code, or system logs) that were not intended to be exposed.
- **Severity**: While LFI primarily leads to **sensitive information disclosure**, it can sometimes be escalated to RCE if the attacker can find a way to place controlled data (like through a log file or an insecure upload function) on the server and then include it.

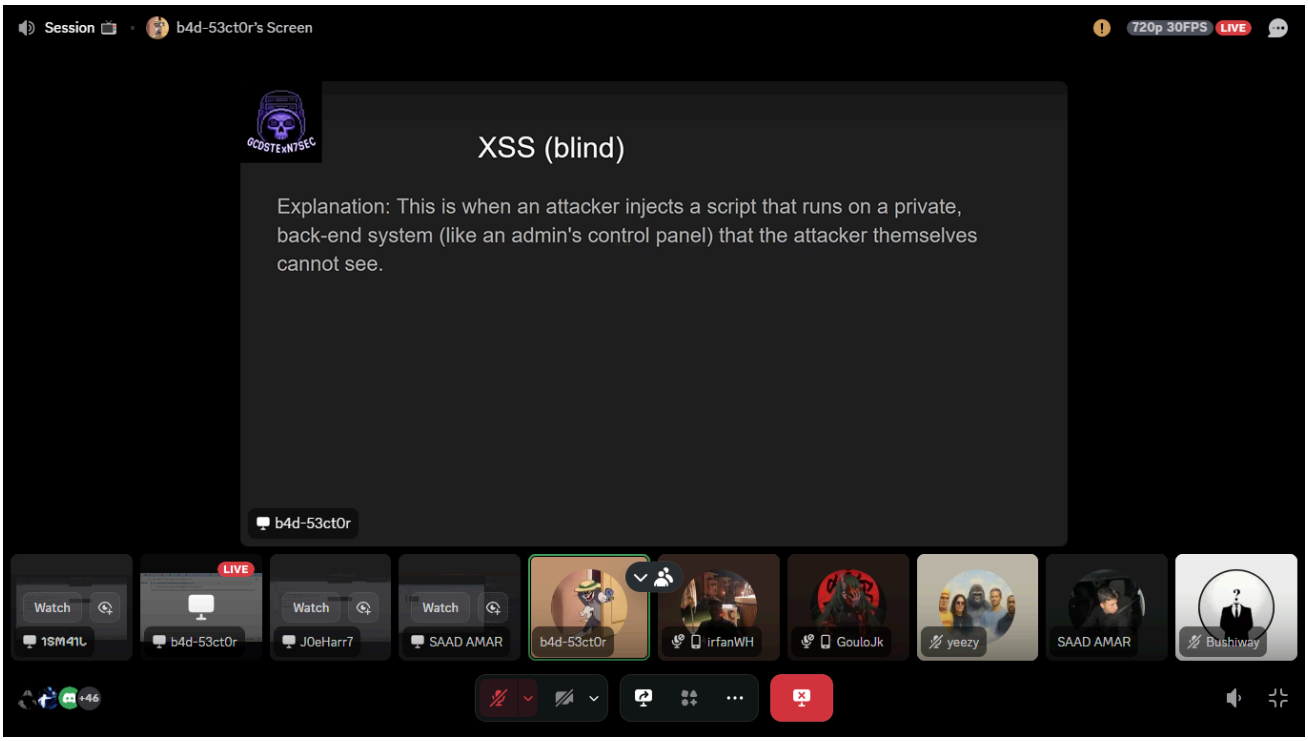
## Cross-site scripting (XSS):



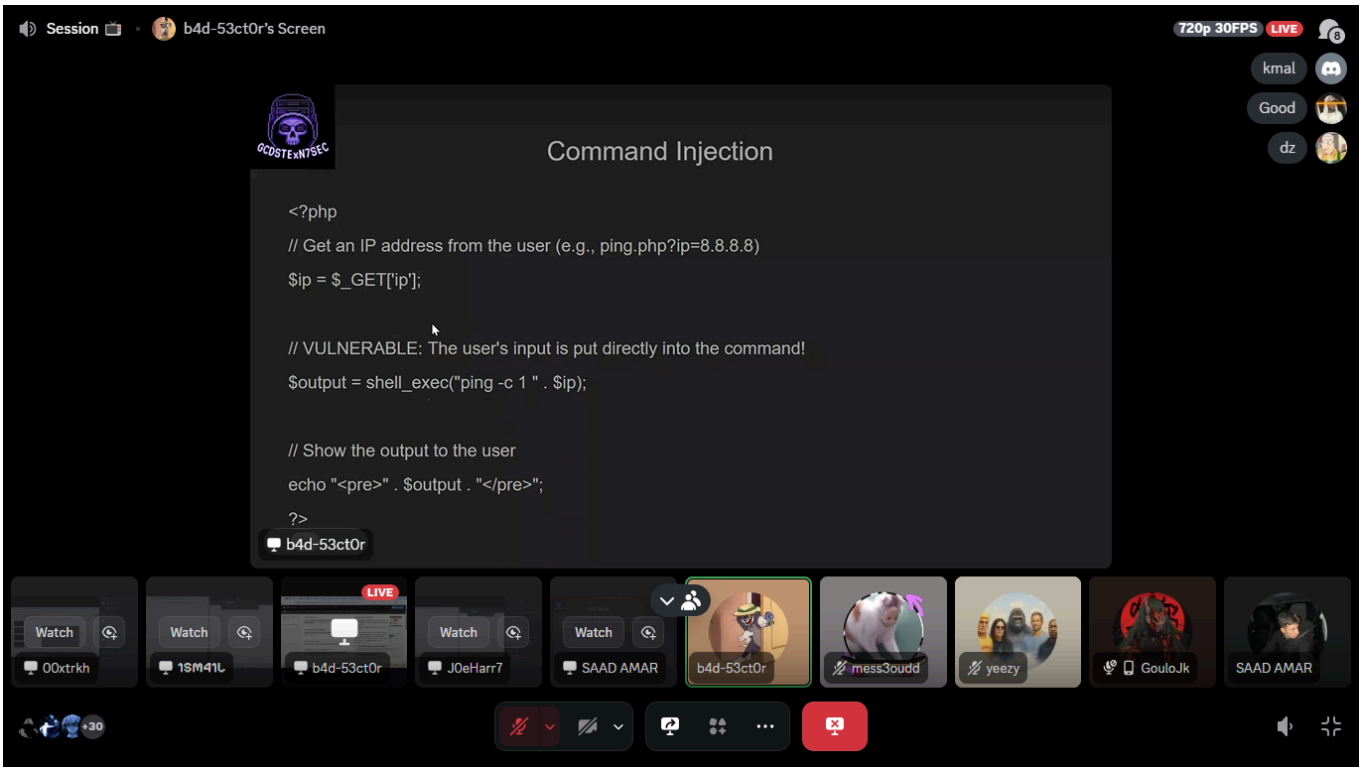
- 


- It can be used to steal cookies and stuff

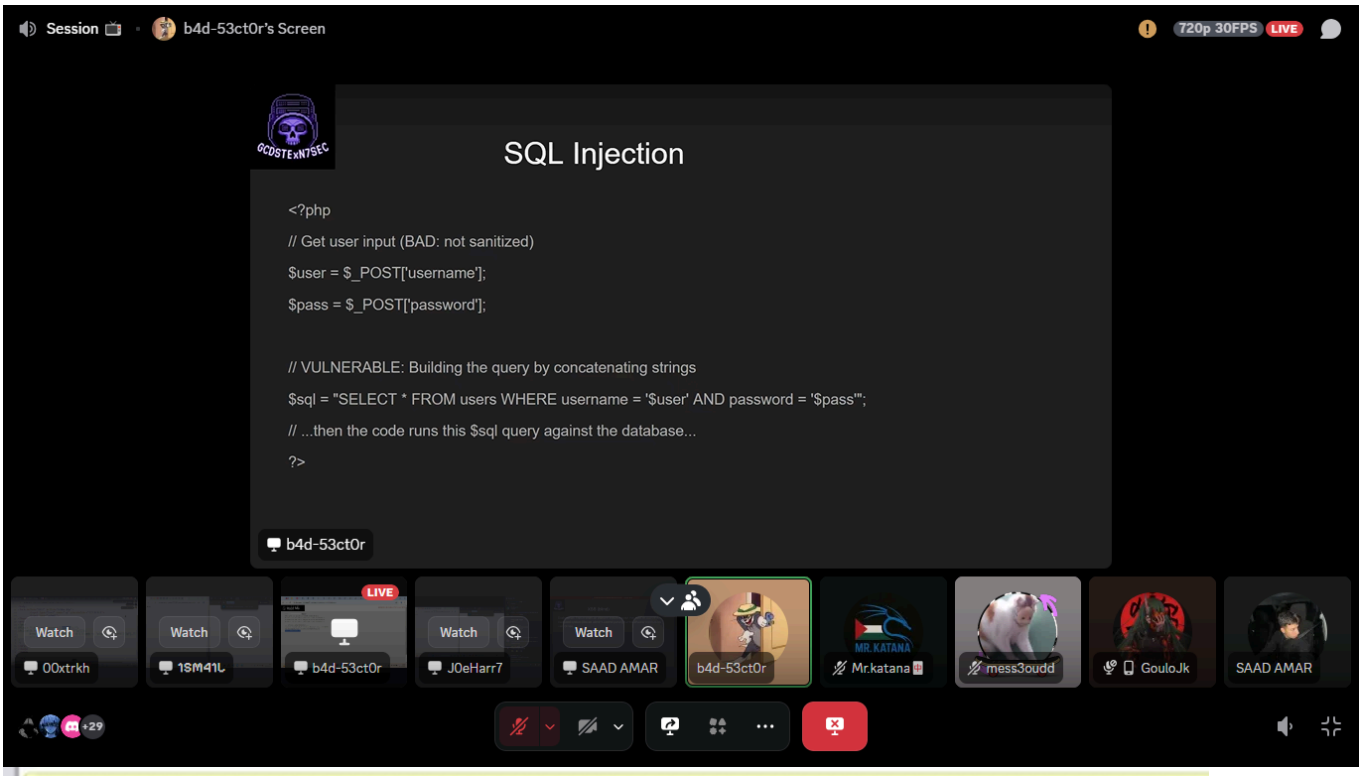
- 

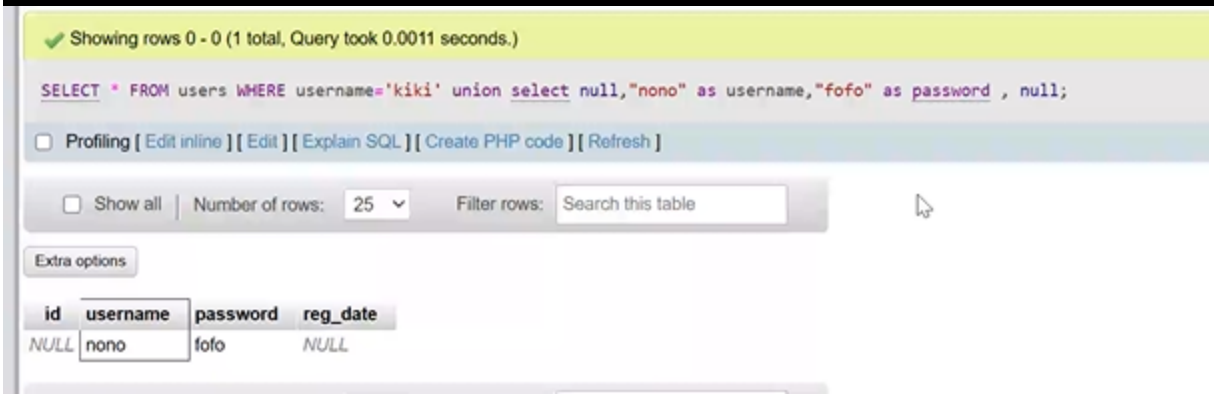



# - Command Injection




# - SQL Injection




- onlinephp.io/password-hash