

iUnite Security

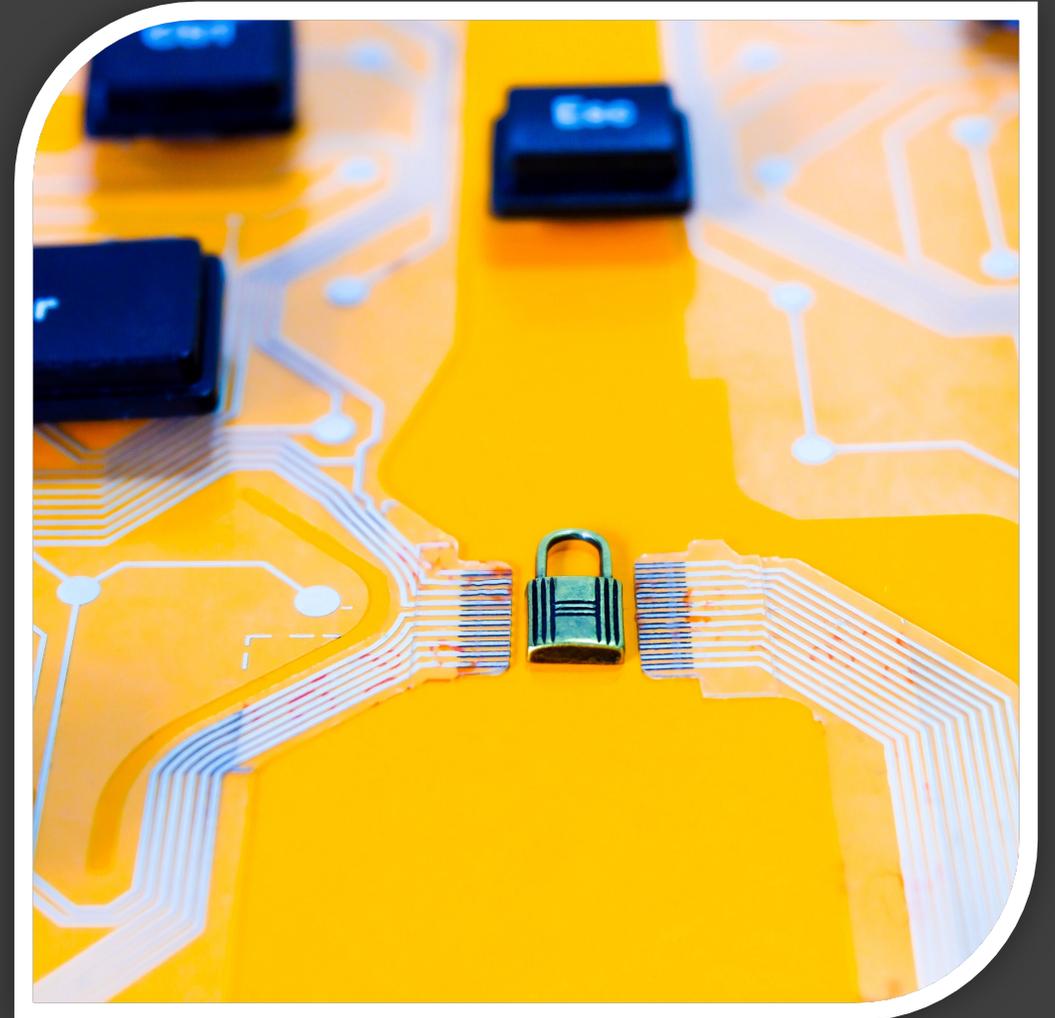
Overview of Security Measures within iUnite



Topic Overview



-  Encryption & Database Security
-  Security Before Page Loads
-  Security During Page Loads
-  Login & Form Security
-  Scans, Alerts, & Monitoring



Encryption & Database Security



What is encryption?



How does it work?



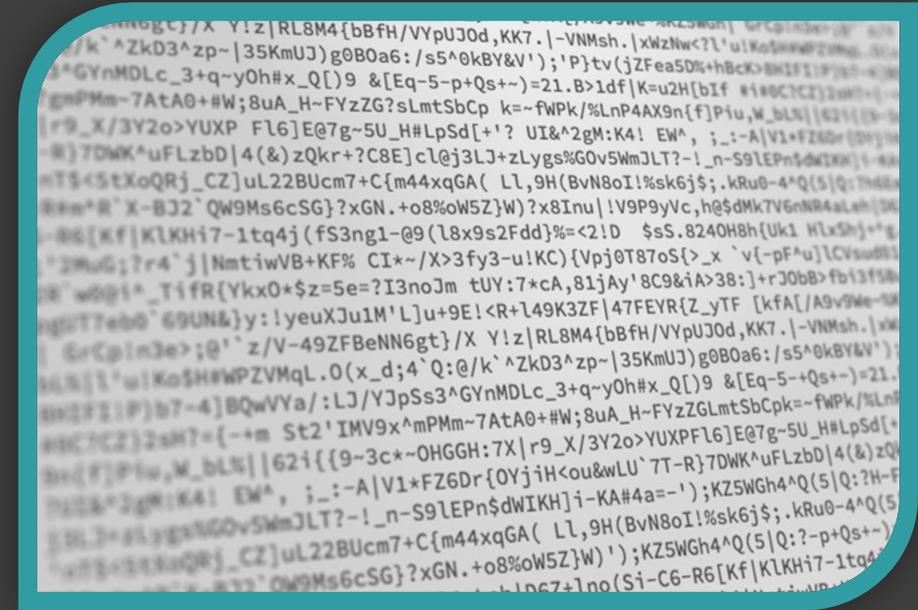
Types of encryption



Encryption at Rest



Minimalistic data access



What is encryption?

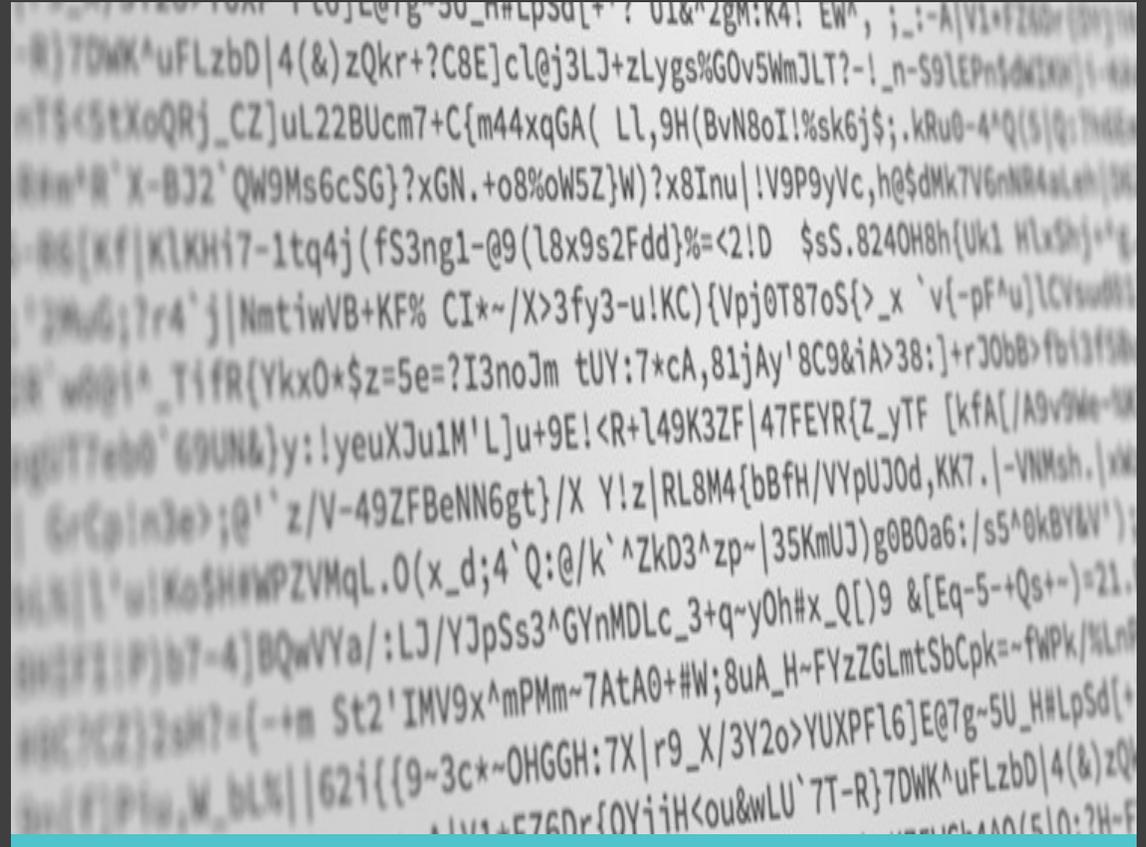
Encryption & Database Security



Encryption is a process that takes readable text and makes it unreadable.

Example:

This is plain text. → dcd5d8dbabe2536a7eed30e6
1474005f3df4d7ba0a921fe3a
12c6b0f62dc8741



How does it work?

Encryption & Database Security



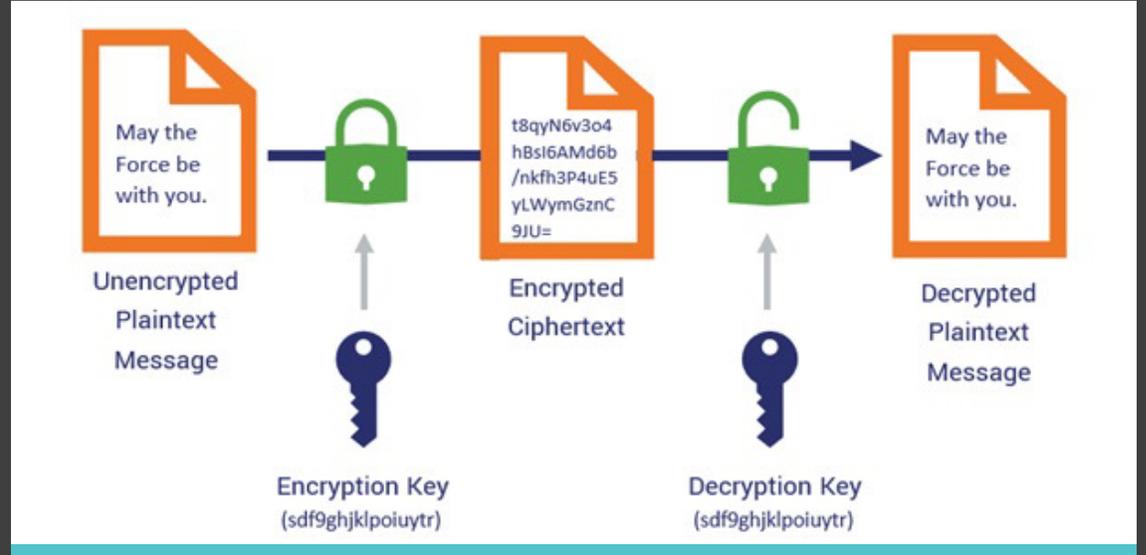
Encryption function turns original data into unreadable “cipher text.”



Cipher text is sent to intended recipient.



Recipient has a “secret” encryption key used to turn the cipher text back into the original data.

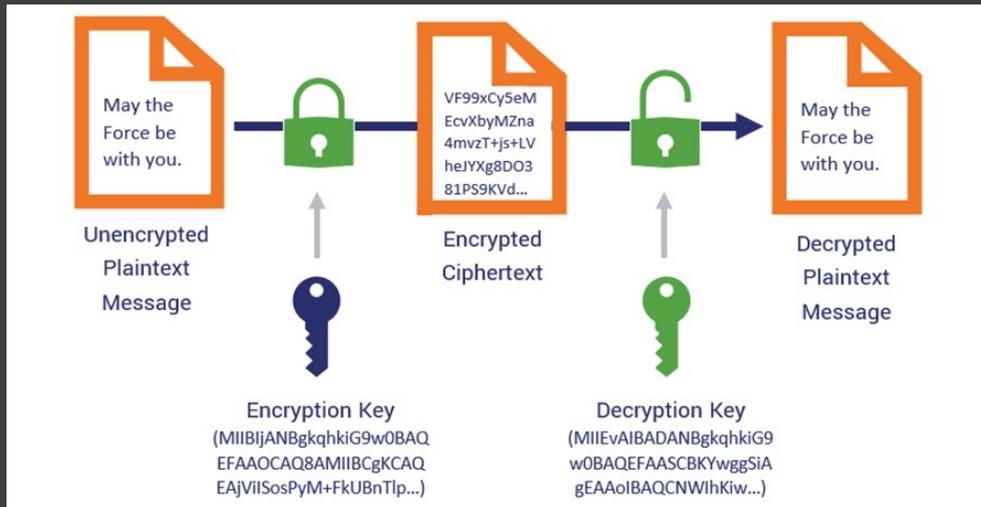


Types of encryption.

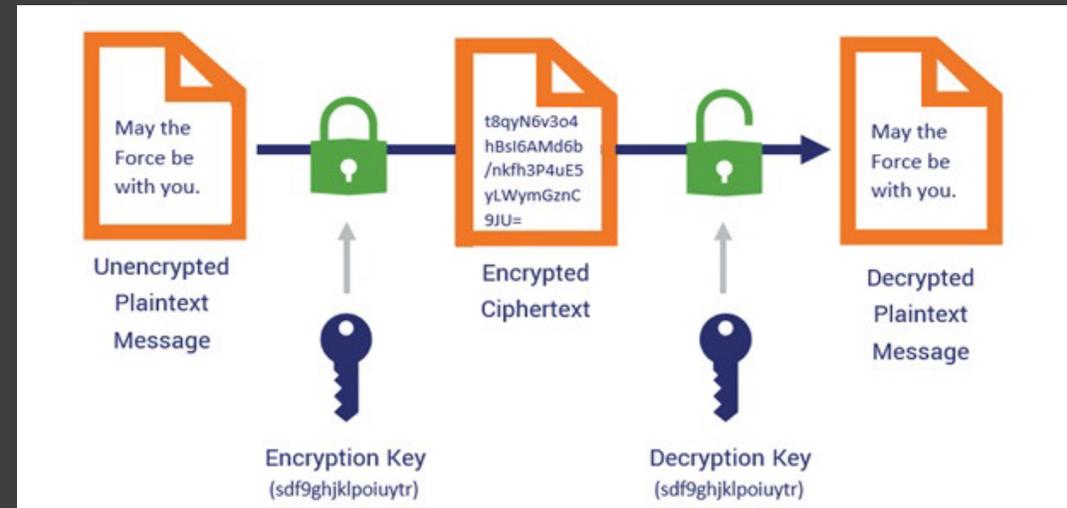
Encryption & Database Security



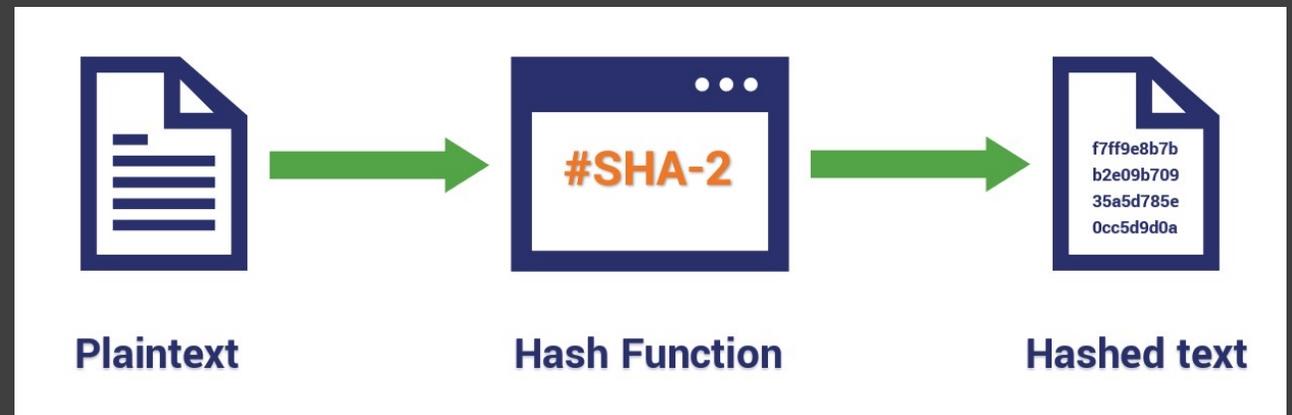
Symmetric Encryption



Hashing



Asymmetric Encryption



Encryption at Rest

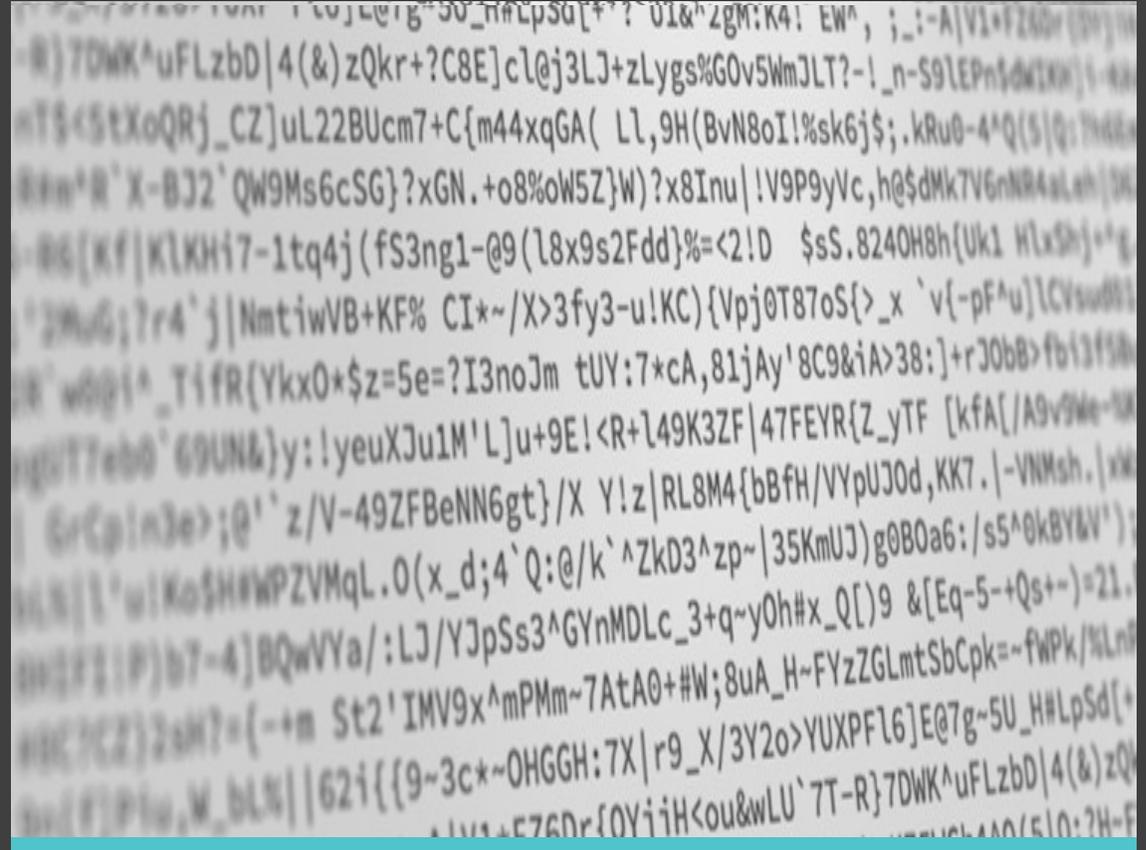
Encryption & Database Security



Data is encrypted as it is stored within the database.

Data is only decrypted as there are authenticated requests.

Prevents physical data theft and digital data theft if the database is accessed outside of iUnite.



Minimalistic Data Access

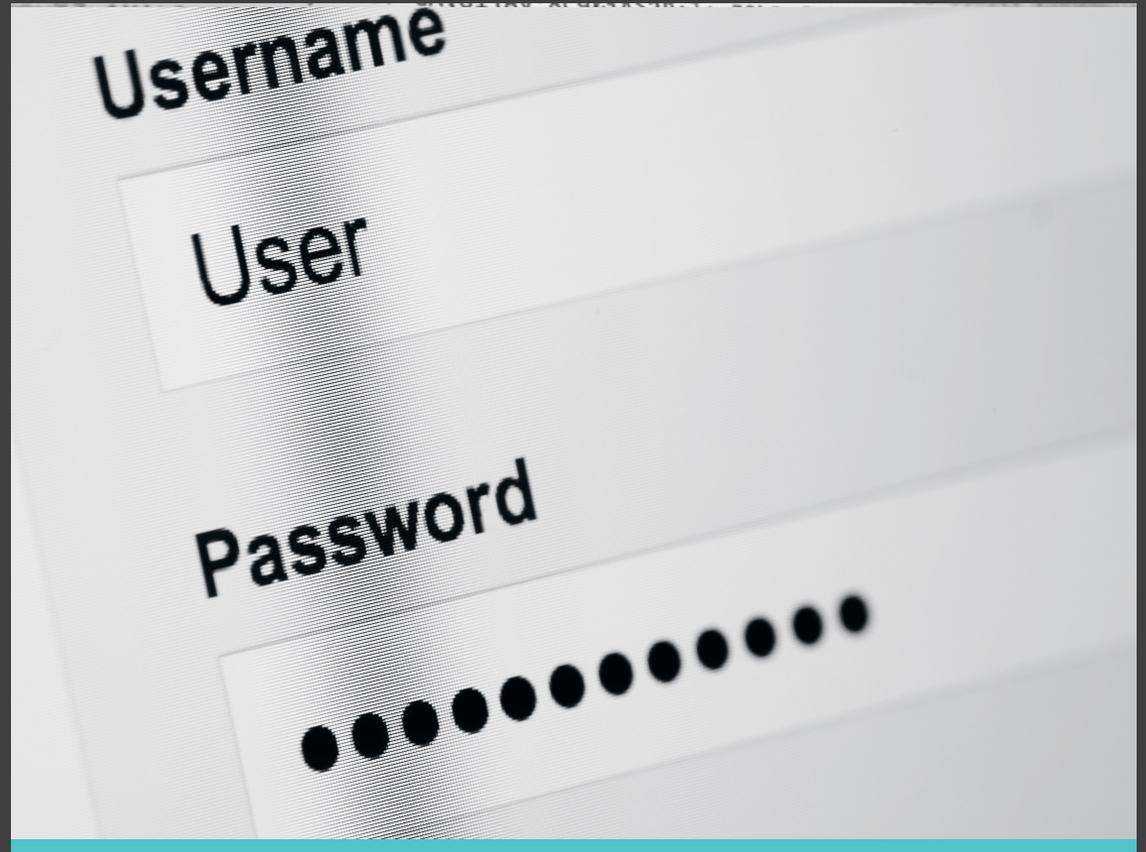
Encryption & Database Security



Database server is separate from the web server.

iUnite is granted access to the database with the fewest number of permissions possible.

Database server can only be accessed with resources in the same account and region.



Security Before Page Loads



IP Address Check



Web Application Firewall (WAF)



DDoS Protection



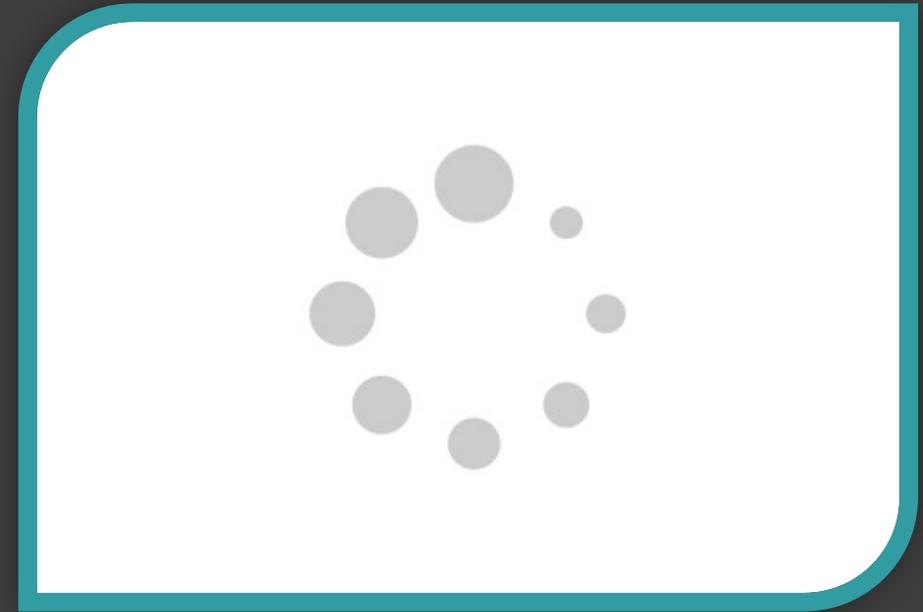
Bot Blocking



Virtual Patching & Hardening



Geo-blocking



IP Address Check

Security Before Page Loads



 All traffic must come through the firewall.

 Attempts to access iUnite.ca by bypassing the firewall will get a 403 error.

 IP is checked against known problem, block listed, and allow listed IP addresses.

Forbidden

You don't have permission to access this resource.

Web Application Firewall (WAF)

Security Before Page Loads



01

DDoS Protection

Dedicated Denial of Service attacks aim to overload a server or otherwise prevent normal function through traffic volume.

02

Bot Blocking

Known malicious or other suspicious bots attempting to access a website are blocked and denied all access.

03

Virtual Patching & Hardening

Attacks based on known exploits of outdated software on a server are prevented in transit to the server.

04

Geo-blocking

Block or allow traffic based on its originating location. iUnite blocks all traffic outside Canada.

Security During Page Loads



Web Application Firewall (internal)



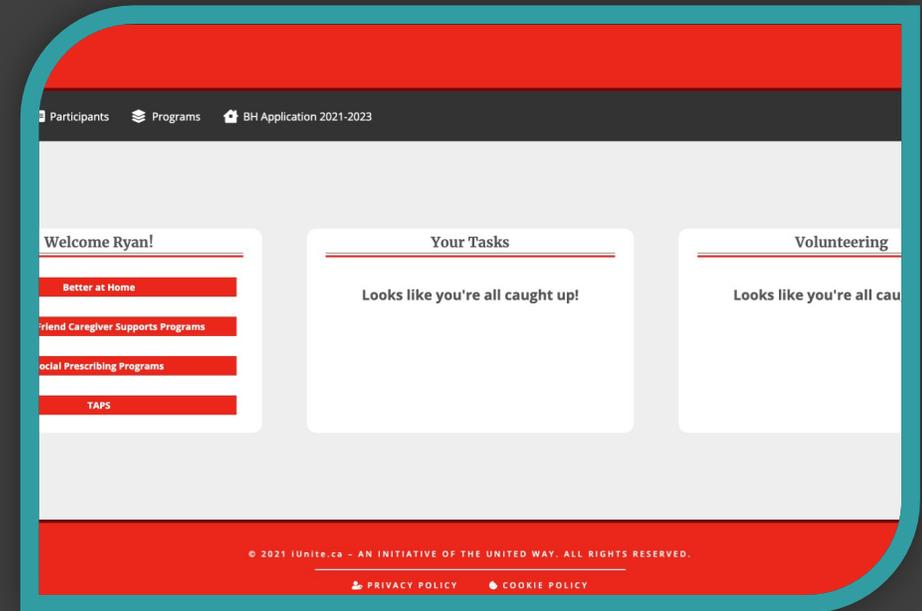
Automatic logout



Access Permissions



Automatic redirects



Web Application Firewall (WAF)

Security During Page Loads



01

DDoS Protection

Dedicated Denial of Service attacks aim to overload a server or otherwise prevent normal function through traffic volume.

02

Bot Blocking

Known malicious or other suspicious bots attempting to access a website are blocked and denied all access.

03

Virtual Patching & Hardening

Attacks based on known exploits of outdated software on a server are prevented in transit to the server.

04

Geo-blocking

Block or allow traffic based on its originating location. iUnite blocks all traffic outside Canada.

Automatic logout

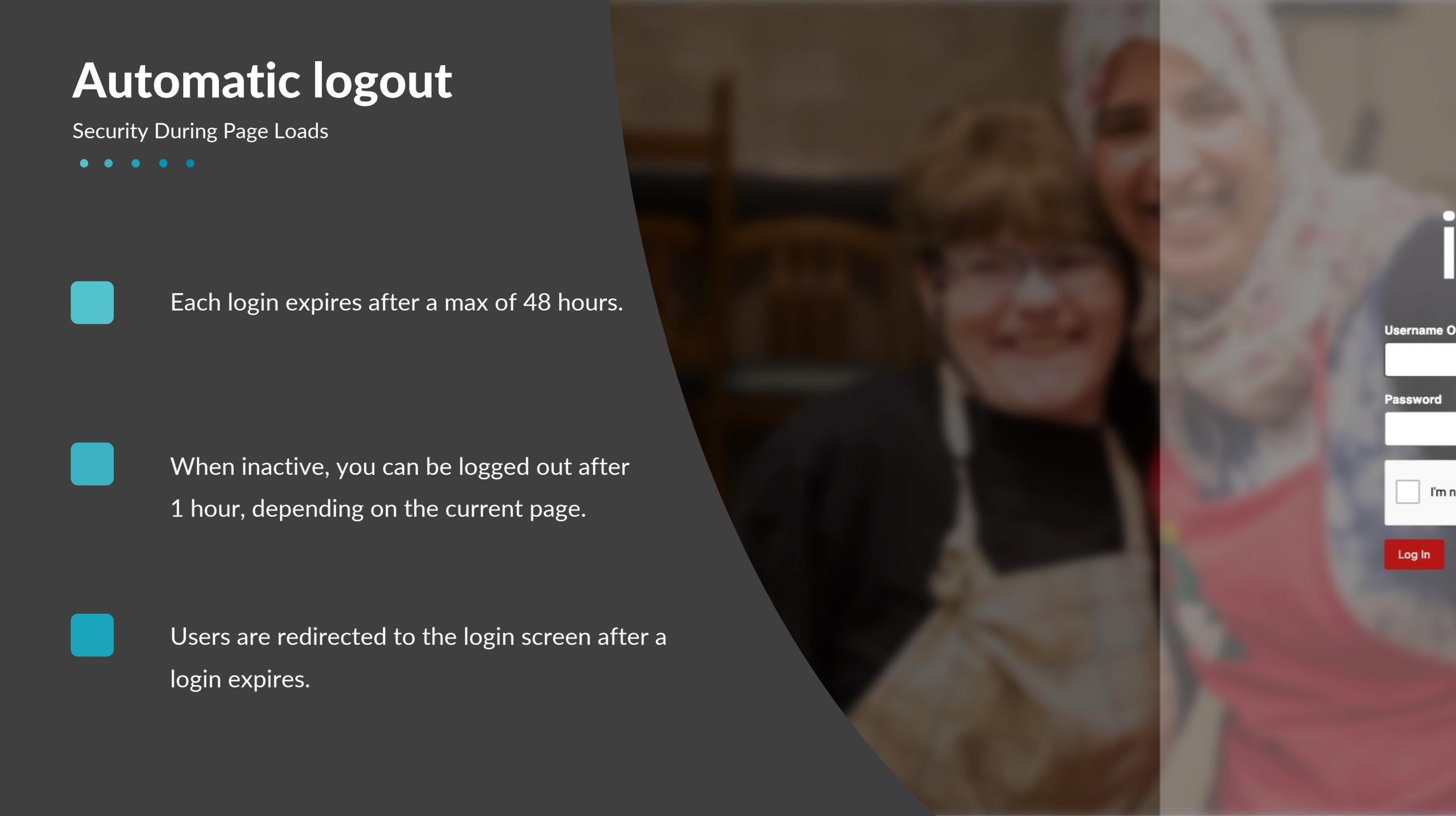
Security During Page Loads



 Each login expires after a max of 48 hours.

 When inactive, you can be logged out after 1 hour, depending on the current page.

 Users are redirected to the login screen after a login expires.





Username

Password

I'm new

Access Permissions

Security During Page Loads



User Role

Check if the user's account has a sufficient role to access the page or data.



Data Access

Check if the user's can view/edit the field or data being displayed.



Page/Form Access

Check if the user can view/edit the current page and/or form being loaded.



DB Table Access

Check if the user can read and/or write from a specific database table.

Automatic redirects

Security During Page Loads



Redirections are based on account permissions.

If still logged in, redirect to homepage.

If logged out, redirect to login page.

Accounts Programs BH Application 2021-2023

Hi Ryan!

Sports Programs

Your Tasks

Looks like you're all caught up!

Login & Form Security



Passwords



Login Obscurity



reCAPTCHA



Two-Factor Authentication (2FA)



Nonces



Account Permissions



Data Sanitization

The screenshot shows a web application interface for 'Participant Intake'. At the top, there is a navigation bar with links for 'Participants', 'Programs', and 'BH Application 2021-2023'. Below the navigation bar, the main heading is 'Participant Intake'. A progress indicator shows 'Step 1 of 5: Participant Info' with a 20% completion bar. The form is divided into sections: 'PART I' with a note '* denotes required fields per UWLM reporting requirements'. Under 'PART I', there are two dropdown menus: 'Programs: *' (with a 'Click to select...' prompt) and 'Participant Region: *' (with a '-- Select Region --' prompt). Below these is a 'Status:' section with radio buttons for 'Active' and 'Inactive'. The 'Participant Information' section is partially visible at the bottom, starting with a 'Name: *' label and a dropdown menu.

Passwords

Login & Form Security



- Must be “Strong” to be allowed.
- Algorithm used looks for patterns, common groupings, and minor modifications to words (i.e. leet speak where some letters are replaced by numbers).
- Not strong enough alone for authentication.

New Password

Password123!

Very weak

Hint: The password should be at least twelve characters long. To make it stronger, use upper and lower case letters, numbers, and symbols like ! " ? \$ % ^ & .)

I'm not a robot

reCAPTCHA
Privacy - Terms

Generate Password

Save Password

[← Return to log in form](#)

Password123!

Very weak

Passw0rd123!

Weak

!Passw0rd123!

Medium

!Pa5w0rd123!

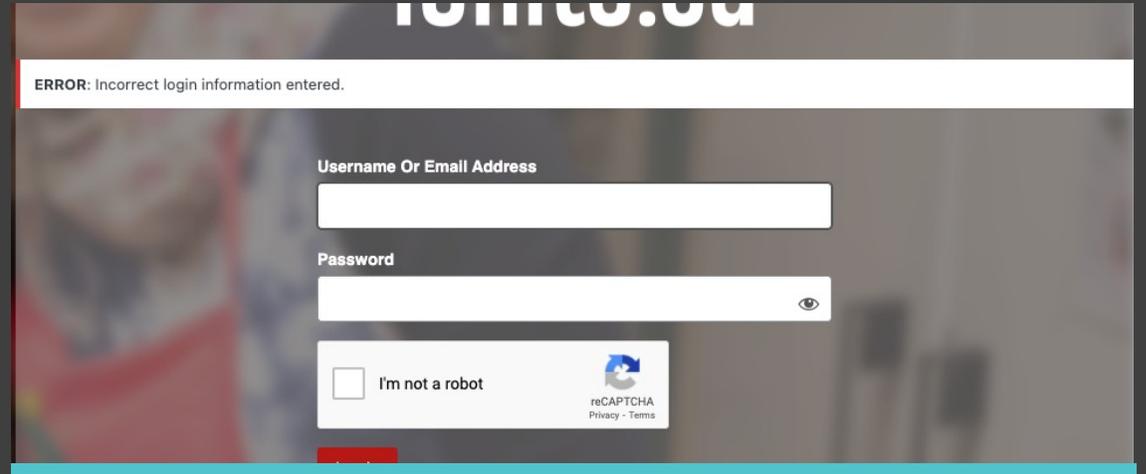
Strong

Login Obscurity

Login & Form Security



- Shows the same error messages regardless if login attempt is a real account or not.
- Password reset does not confirm if an email was sent or not, to avoid confirming account emails or usernames.
- Makes it harder to hack a password if guessing at emails and usernames.



reCAPTCHA

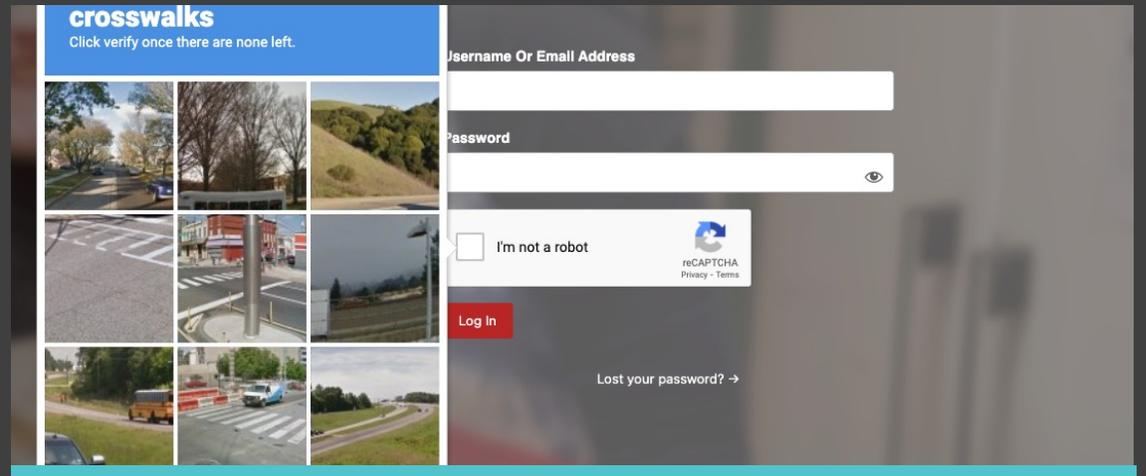
Login & Form Security



Two types: invisible and checkbox

Assesses how pieces of a website are regularly used to weed out suspicious users and bots.

Based on how user behaves on a website, a risk score between 0 (bad traffic) and 1 (good traffic) is created.



Two-Factor Authentication (2FA)

Login & Form Security



Email code sent is unique for that account and time sensitive (5 minutes).

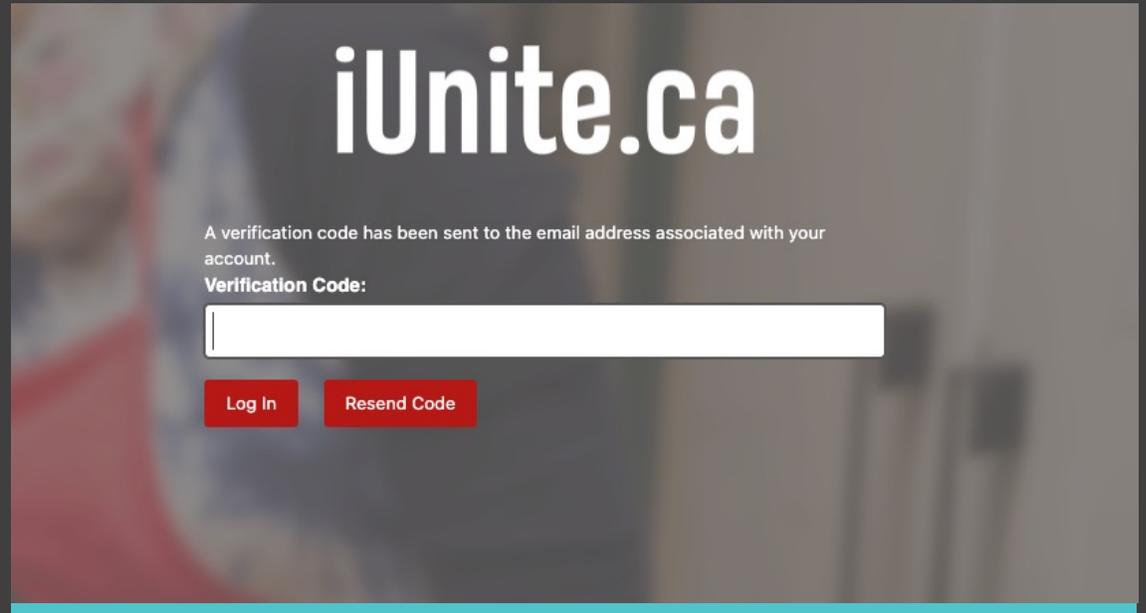


To hack into an account, hacker would need to have a username or email, current password, and access to the email account.



Future options:

- SMS/Text
- Time Based One-Time Password (TOTP)
- One-Touch Sign-on
- Magic Link
- Single Sign On (SSO)



Nonces

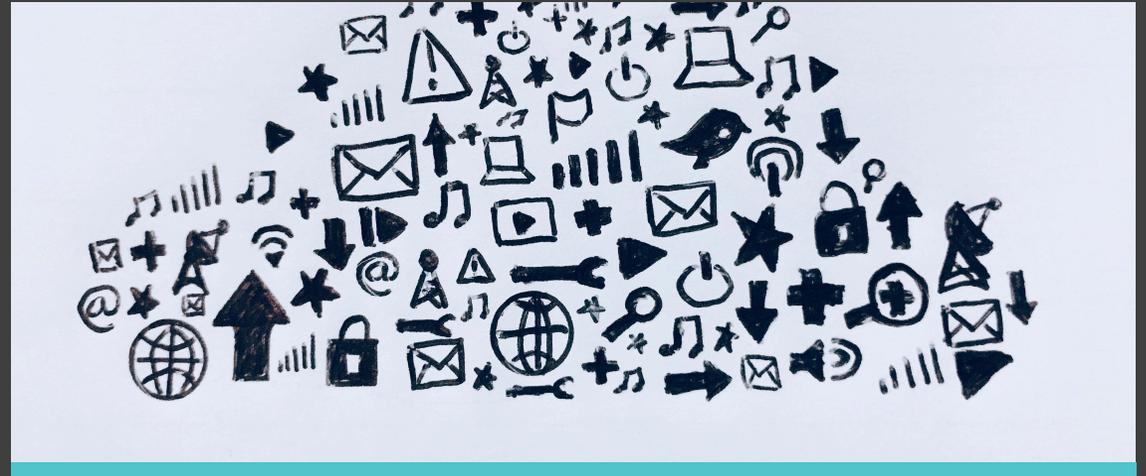
Login & Form Security



Number used once to verify the form submission is from the same user that loaded the page.

Nonces is a grouping of random characters randomly generated (i.e. 39a650fbec)

Same nonce could possibly be used twice or more eventually but would not be the same for multiple accounts at the same time (holds for 12 hours).



Access Permissions

Login & Form Security



User Role

Check if the user's account has a sufficient role to access the page or data.



Data Access

Check if the user's can view/edit the field or data being displayed.



Page/Form Access

Check if the user can view/edit the current page and/or form being loaded.



DB Table Access

Check if the user can read and/or write from a specific database table.

Data Sanitization

Login & Form Security



Converts all submitted data to strings (text).

Checks for and removes code snippets. These are typically in the format of:

```
<script> var code = something; </script>
```

Files are scanned during upload for potential malicious code, suspicious file data, and file type mismatches (i.e. uploaded as a .pdf, but is really a .exe file).



Scans, Alerts, & Monitoring



Scans



Alerts

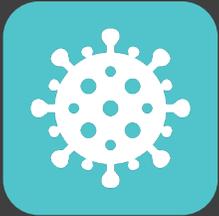


Monitoring

The screenshot shows a web application interface for 'Participant Intake'. At the top, there is a navigation bar with 'Participants', 'Programs', and 'BH Application 2021-2023'. The main heading is 'Participant Intake'. Below the heading, there is a progress indicator for 'Step 1 of 5: Participant Info' with a 20% completion bar. The form is divided into sections: 'PART I' with a note '* denotes required fields per UWLM reporting requirements'. It includes fields for 'Programs: *' (a dropdown menu), 'Participant Region:' (a dropdown menu), and 'Status:' with radio buttons for 'Active' and 'Inactive'. Below this is a section for 'Participant Information' with a 'Name: *' field and several other input fields.

Scans

Scans, Alerts, & Monitoring



Malware & Hack Scanning

Looks for malicious code and unusual content within files and the database.



DNS Scanning

Checks for any changes to the domain's DNS records. Changes here can change where the domain goes.



File integrity

Checks for file changes against previous scans and/or repositories.

Alerts

Scans, Alerts, & Monitoring



User account changes (add, edit, delete, failed logins)



Website changes (software activated/deactivated, odd files, file changes, DNS changes)



Security events (blocked actions, unauthorized form submissions, blocked users/traffic)



Monitoring

Scans, Alerts, & Monitoring



Blacklist monitoring



Security Logging:

- Logins
- Failed login attempts
- Form submissions/Actions taken
- User account changes (add/edit/delete)
- Password reset attempts



Questions



If you have any additional questions after this webinar, please contact us.



help@iunite.ca

Login

Username/Email:

Password:

Login

[← Forgot Password](#)

[Sign Up →](#)