Vault Raider: Stealthy UI-based Attacks Against Password Managers in Desktop Environments

Andrea Infantino, Mir Masood Ali, **Kostas Solomos** and Jason Polakis

{ainfan5, mali92, polakis}@uic.edu
solomos@brandeis.edu

NDSS SYMPOSIUM/2026

UNIVERSITY OF ILLINOIS CHICAGO

Brandeis UNIVERSITY

Password Managers

- Widely adopted by millions of users and enterprises

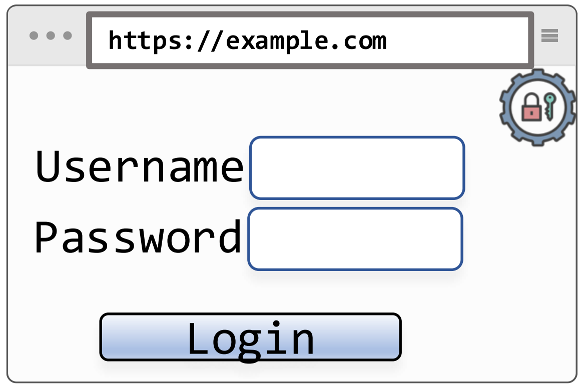
- Store and manage users’ credentials
  - Account and system credentials
  - OTP codes, payment data, keys

- Provide automatic *credential autofill* across applications

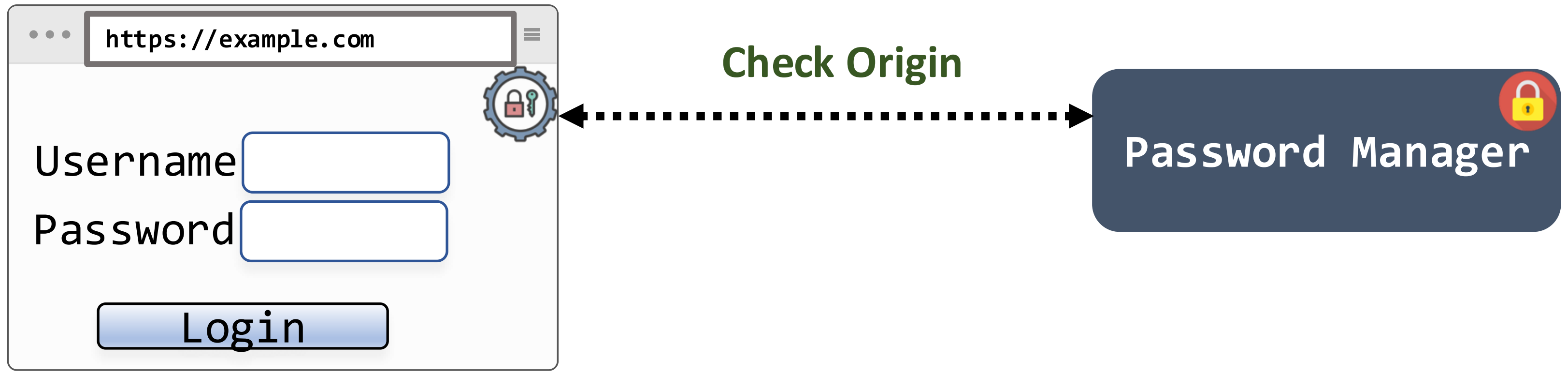
Desktop-based Password Managers

- Password managers were originally designed for web browsers
- Users now authenticate in *native desktop applications*
    - Zoom, Slack, Discord, Teams, . . .
- Autofill expanded to system-wide desktop integration across OSes
- Desktop autofill operates under a ***different security model*** than the browser

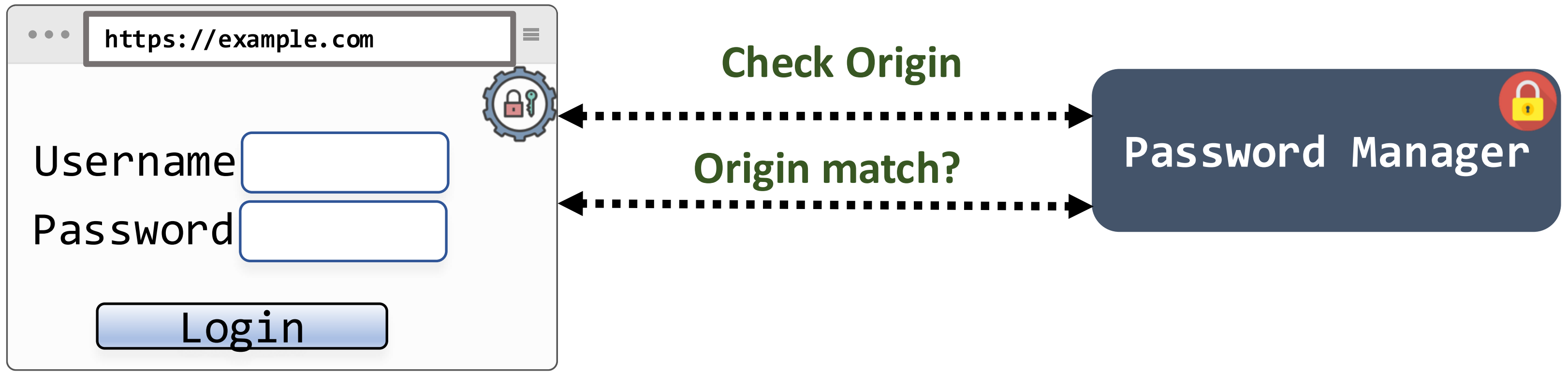
Autofill Model: Browser vs Desktop



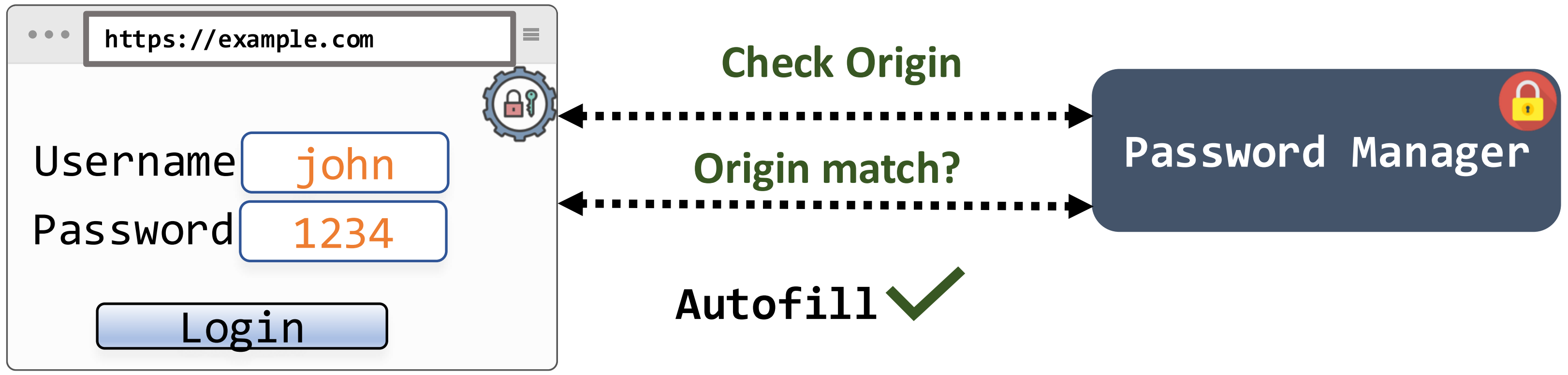
Autofill Model: Browser vs Desktop



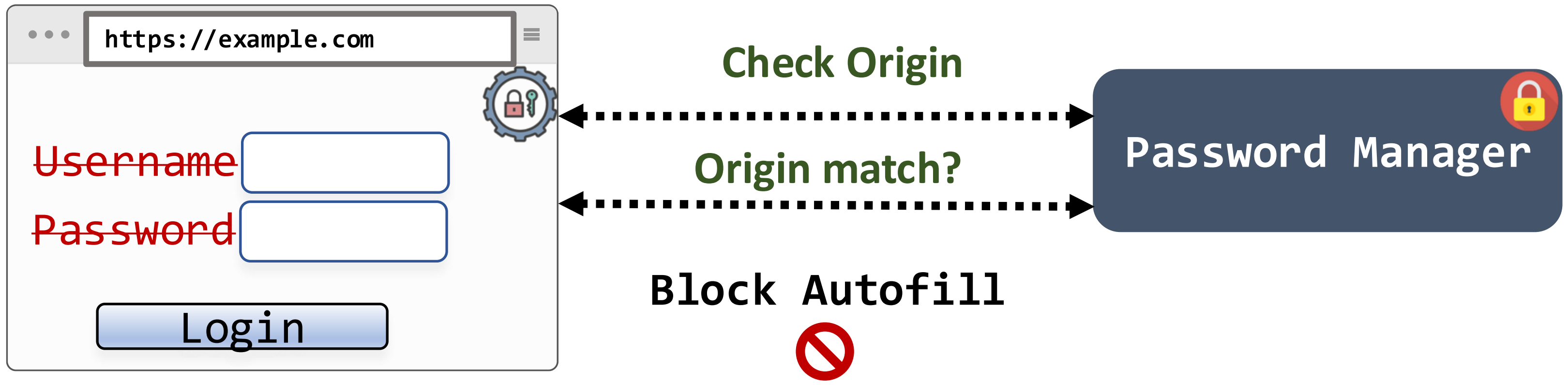
Autofill Model: Browser vs Desktop



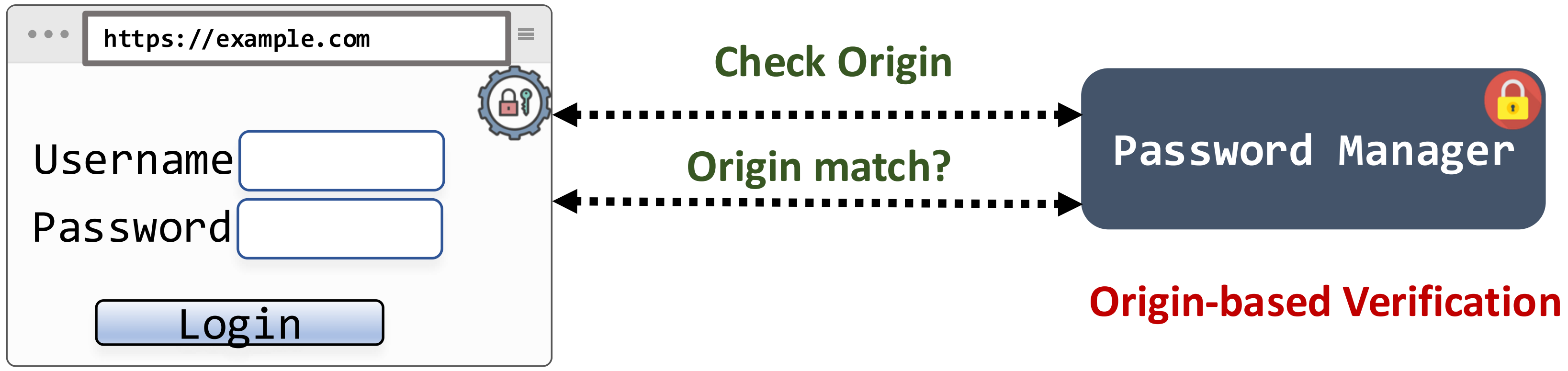
Autofill Model: Browser vs Desktop



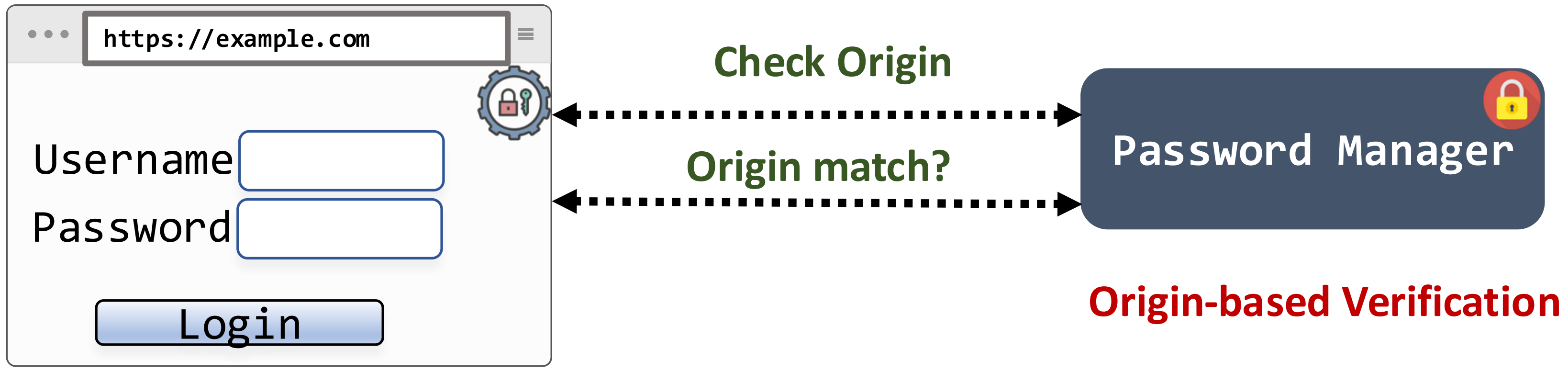
Autofill Model: Browser vs Desktop



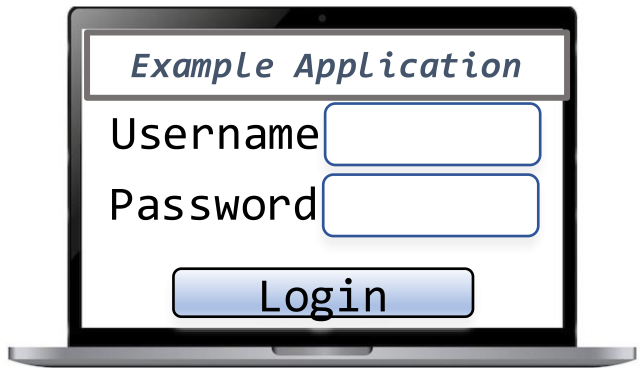
Autofill Model: Browser vs Desktop



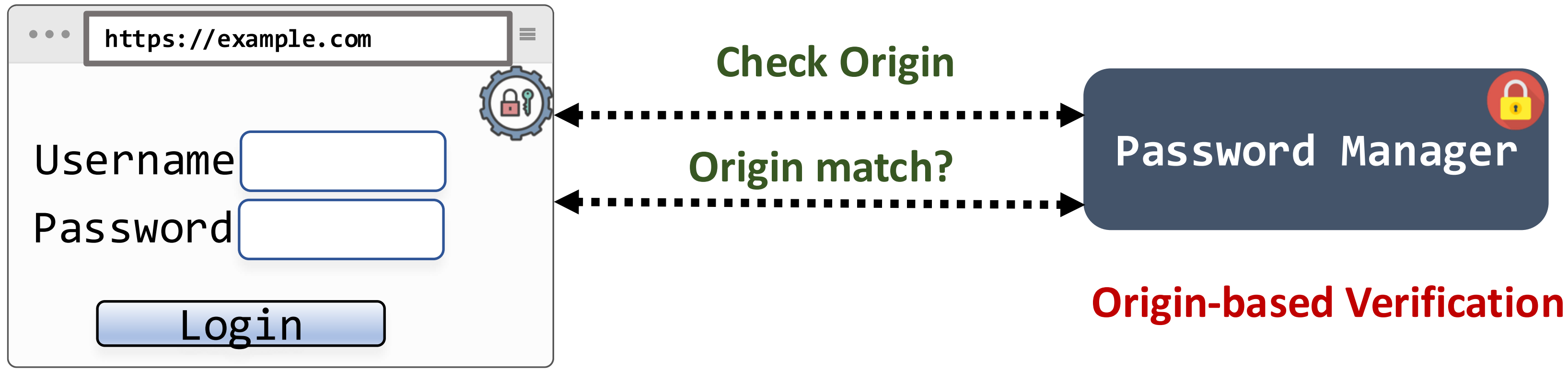
Autofill Model: Browser vs Desktop



Origin-based Verification

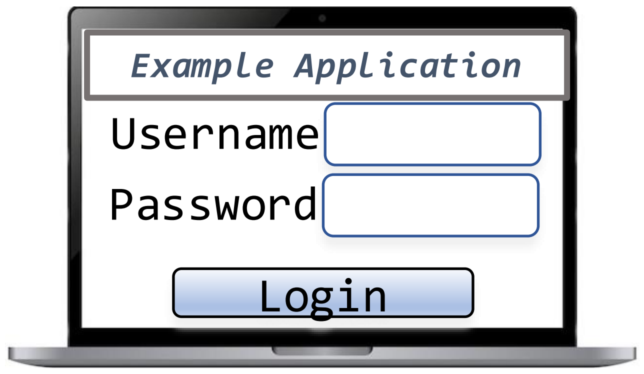


Autofill Model: Browser vs Desktop

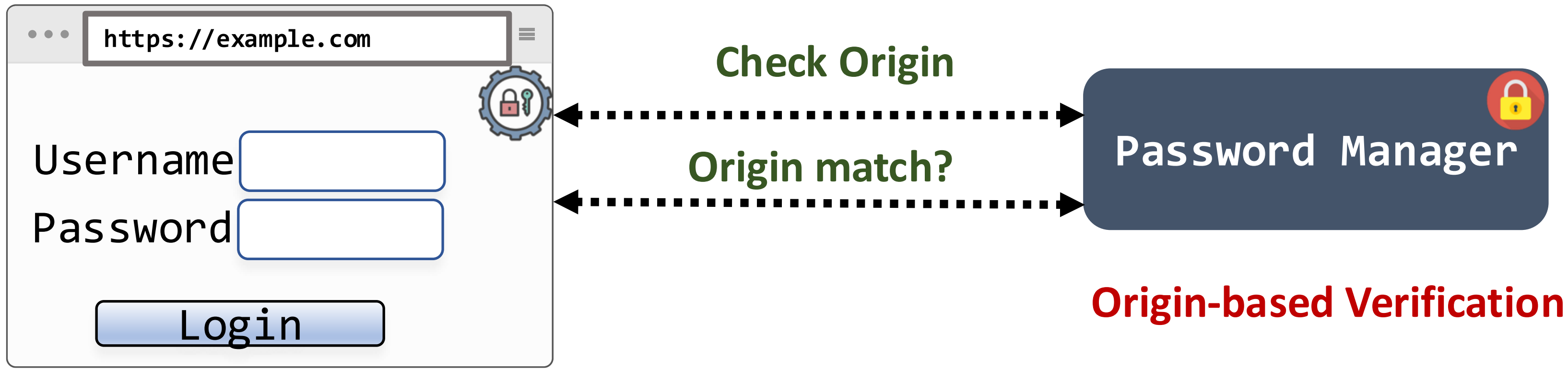


Origin-based Verification

No explicit URL

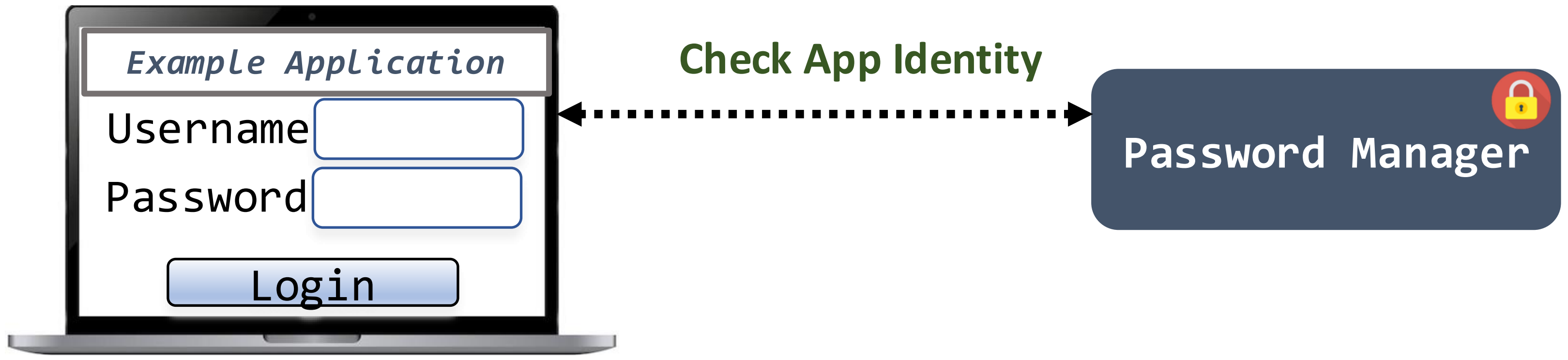


Autofill Model: Browser vs Desktop

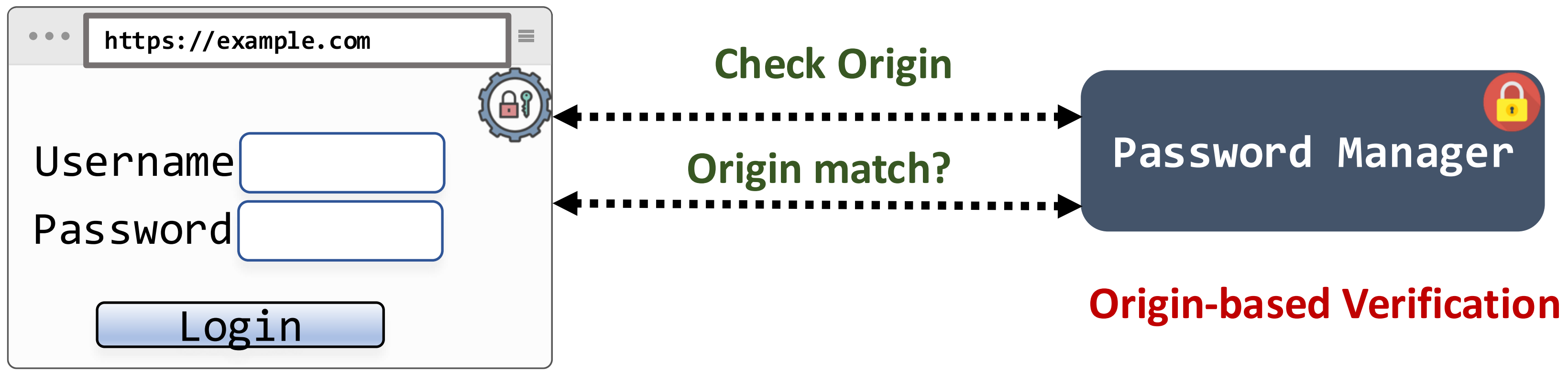


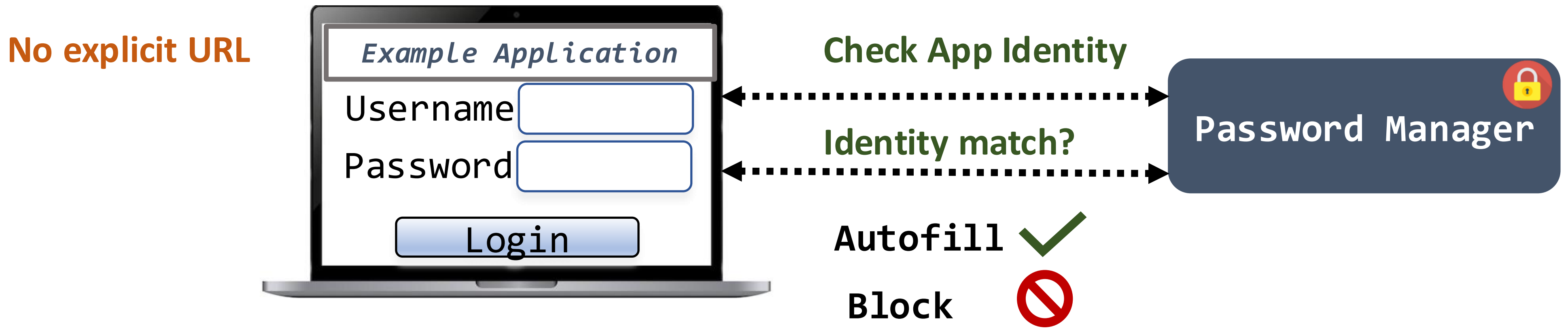
Origin-based Verification

No explicit URL

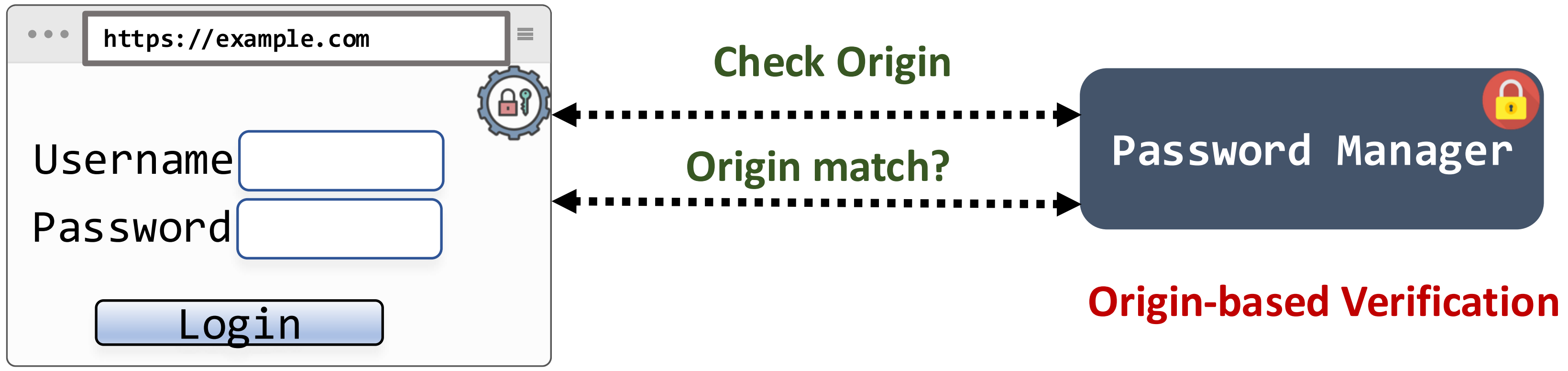


Autofill Model: Browser vs Desktop

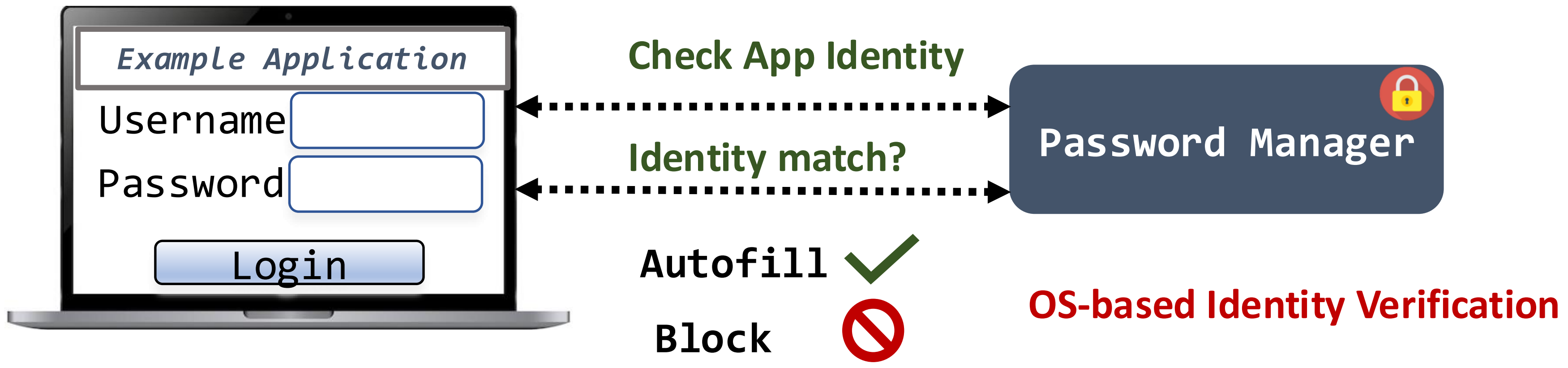




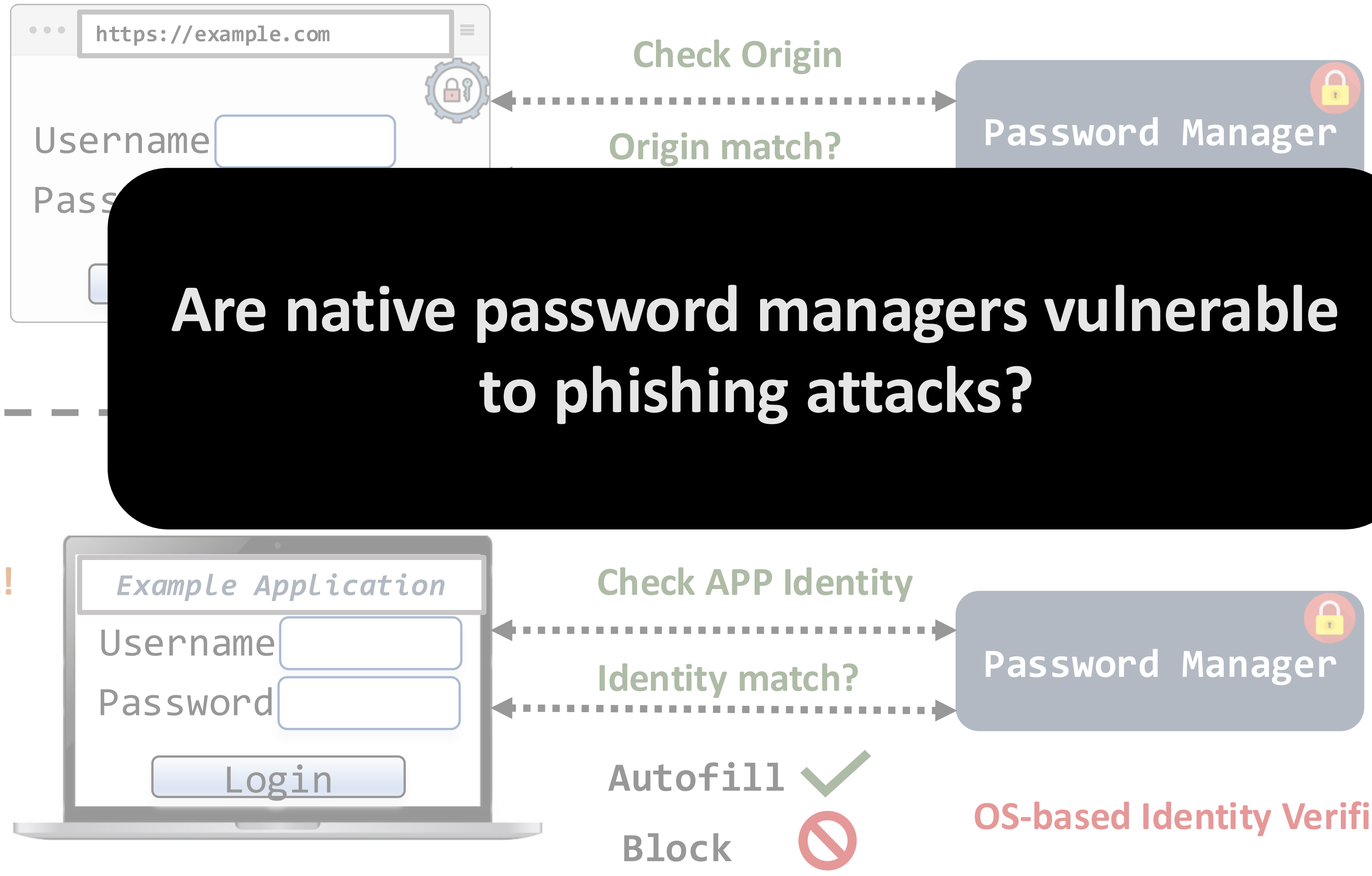
Autofill Model: Browser vs Desktop



No explicit URL



Autofill Model: Browser vs Desktop



Are native password managers vulnerable to phishing attacks?

Threat Model

- Attacker controls a malicious native application running with standard user privileges (*phishing app*)

- Attacker capabilities
  - Control appearance and metadata to impersonate a legitimate application
  - Ability to trigger password manager autofill through intended OS mechanisms
  - *No OS compromise or elevated privileges required*

- **Bypass application-level verification and inject credentials into attacker-controlled application**

OS-Level Identity Verification

macOS Application Identity Model

- Applications are distributed as signed *app bundles*
  - **Bundle Identifier**: immutable OS-level identity
  - **Display Name**: user-facing label
  - **Code Signature**: cryptographic developer identity

- Autofill workflow (*1Password*)

  1. Autofill invoked
  2. Password manager retrieves the target *Bundle ID*
  3. Verification:
     - Valid code signature
     - Trusted developer
     - Stored application binding
  4. If validation succeeds, inject credentials

1Password Quick Access Feature

- Secondary interface for simplified autofill and credential management
- Allows manual selection of stored credentials
- Allows credential transfer into the currently focused application

1Password Quick Access Feature

- Secondary interface for simplified autofill and credential management
- Allows manual selection of stored credentials
- Allows credential transfer into the currently focused application
- **Does not verify the target application identity**

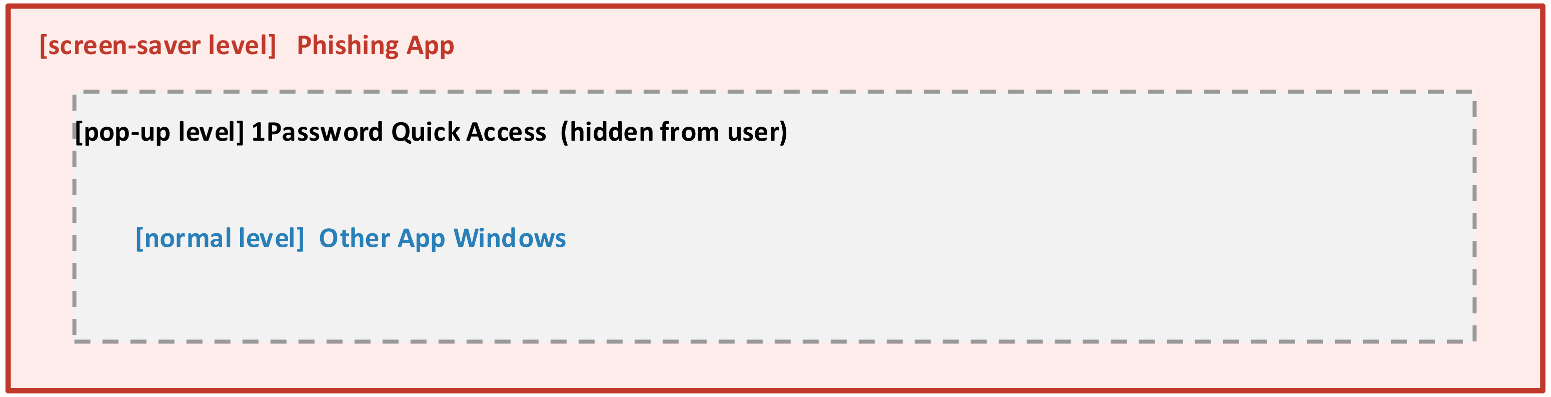
1Password Quick Access: Attack Workflow

1Password Quick Access: Attack Workflow

- **Activation**
  1. *Synthetic* shortcut triggers Quick Access
  2. Quick Access window opens

1Password Quick Access: Attack Workflow

- **Activation**
  1. *Synthetic* shortcut triggers Quick Access
  2. Quick Access window opens
- **Concealment**
  3. Attacker window is *elevated* to screen-saver level


[screen-saver level] Phishing App
[pop-up level] 1Password Quick Access (hidden from user)
[normal level] Other App Windows

1Password Quick Access: Attack Workflow

- **Activation**

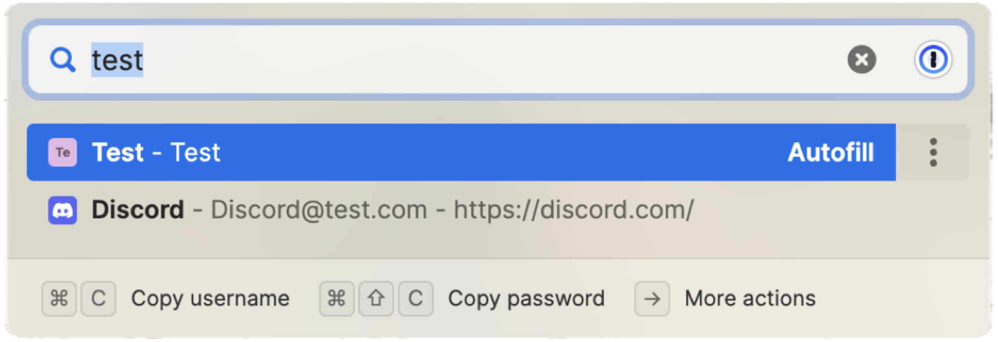
  1. *Synthetic* shortcut triggers Quick Access
  2. Quick Access window opens

- **Concealment**

  3. Attacker window is *elevated* to screen-saver level

- **Credential Extraction**

  4. Programmatically interacts with the vault and selects target credentials



1Password Quick Access: Attack Workflow

- **Activation**

  1. *Synthetic* shortcut triggers Quick Access
  2. Quick Access window opens

- **Concealment**

  3. Attacker window is *elevated* to screen-saver level

- **Credential Extraction**

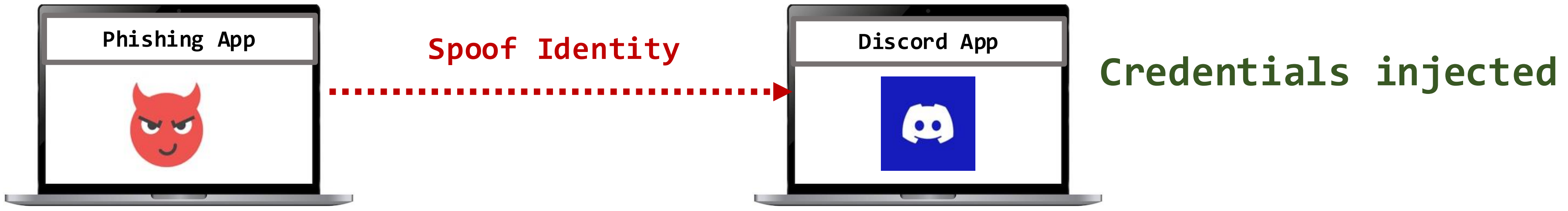
  4. Programmatically interacts with the vault and selects target credentials
  5. Credentials injected into the *phishing app's hidden fields*

1Password: Attack Impact

- **System password harvesting**
  - Predictable system password records (e.g., `Bob’s Macbook Pro`)
  - Reconstruct record name via device metadata
  - Retrieve *system password* (*sudo*)
  - Allows execution of **privileged commands**
- **Vault replication**
  - Attacker triggers 1Password CLI
  - Access auto-generated 1Password Account record
  - Retrieves *Master Password* and *Secret Key*
  - **Synchronize vault** to attacker-controlled device

Additional Password Managers

- Autofill verification mechanisms
  - **Keeper / KeePassXC:** Display Name
  - **MacPass:** Window title
- Identity verification relies on *mutable* application metadata
  - No cryptographic or OS-based validation of the target application
  - Attacker *impersonates* a legitimate application

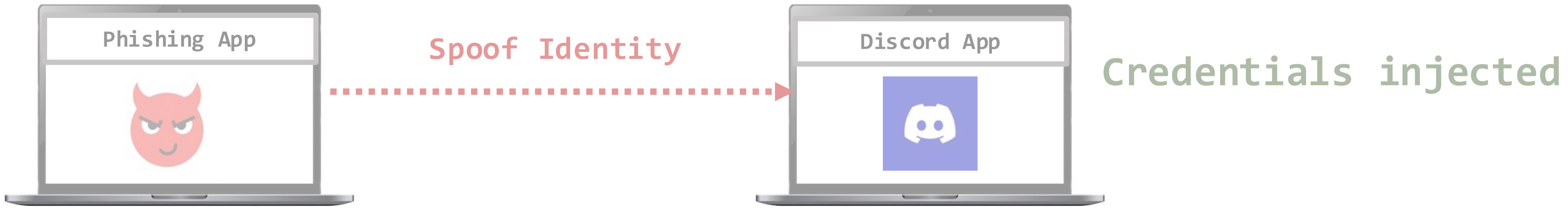


Additional Password Managers

- Autofill verification mechanisms
  - **Keeper / KeePassXC:** Display Name
  - **MacPass:** Window title

OS-level protections are ineffective without secure application authentication

  - No cryptographic or OS-based validation of the target application
  - Attacker *impersonates* a legitimate application



Windows Password Managers

- Security Model
  - No OS-enforced immutable application identity
  - No mandatory code-signing validation
  - UI automation allowed across processes

Windows Password Managers

- Security Model
  - No OS-enforced immutable application identity
  - No mandatory code-signing validation
  - UI automation allowed across processes

- Autofill Target Verification
  - **1Password: No application identity check!**
  - **Keeper / LastPass / KeePassXC:** Window-title comparison (*spoofable*)

Windows Password Managers

- Security Model
  - No OS-enforced immutable application identity
  - No mandatory code-signing validation
  - UI automation allowed across processes

- Autofill Target Verification
  - **1Password: No application identity check!**
  - **Keeper / LastPass / KeePassXC:** Window-title comparison (*spoofable*)

Absence of OS-level application identity prevents reliable autofill validation

Cross-Platform Attack Evaluation

Password Manager	Harvested Credentials	Supported Platforms	Stealthiness	Performance
1Password	👤 🔒 📱 💳	🪟	● ●	11s
Keeper	👤 🔒 📱 💳	🪟	● ○	4s
KeepassXC	👤 🔒 📱		●	17s
MacPass	👤 🔒 📱		●	5s
LastPass	👤 🔒 📱 💳	🪟	●	5s

Cross-Platform Attack Evaluation

Password Manager	Harvested Credentials	Supported Platforms	Stealthiness	Performance
1Password	👤 🔒 📱 💳	⊞	● ●	11s
Keeper	👤 🔒 📱 💳	⊞	● ◐	4s
KeepassXC	👤 🔒 📱		●	17s
MacPass	👤 🔒 📱		●	5s
LastPass	👤 🔒 📱 💳	⊞	●	5s

Proposed Countermeasures

- **Password Managers**
  - Enforce uniform identity validation across all autofill features
  - Bind credential injection to validated applications only
  - Require trusted user interaction as form of authentication and reject synthetic interactions (*prototype available!)*
- **Operating Systems**
  - Prevent layering and concealment of **security-critical** and authorization windows
  - Windows should enforce a protected UI context for secure autofill (similar to macOS)

Disclosure & Mitigations

- **Coordinated Disclosure**
  - All affected password managers notified with technical details and reproduction steps
  - Vulnerabilities reproduced and acknowledged

- **Vendor Responses**
  - Keeper & MacPass: vulnerabilities patched
  - Keeper: bug bounty awarded
  - 1Password: *platform limitations constrain certain defenses against phishing attacks*

Takeaways

Artifacts

- Universal desktop autofill expands the **attack surface** when application identity is not consistently enforced
- We demonstrate stealthy UI-based phishing attacks that harvest credentials, financial info, and bypass 2FA across **all major password managers**
- Secure desktop autofill requires OS-enforced identity checks and protected UI contexts
- Our work strengthens ongoing efforts to improve the security and trustworthiness of desktop password managers

Vault Raider: Stealthy UI-based Attacks Against Password Managers in Desktop Environments

Andrea Infantino, Mir Masood Ali, **Kostas Solomos** and Jason Polakis

{ainfan5, mali92, polakis}@uic.edu
solomos@brandeis.edu

NDSS SYMPOSIUM/2026

UIC UNIVERSITY OF ILLINOIS CHICAGO

Brandeis UNIVERSITY