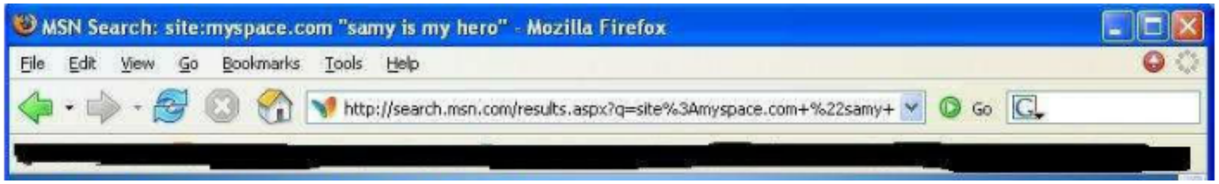
PathCutter: Severing the Self-Propagation Path of XSS JavaScript Worms in Social Web Networks

Yinzhi Cao[§], Vinod Yegneswaran[†], Phillip Porras[†], and Yan Chen[§]

[§]Northwestern Lab for Internet and Security Technology, Northwestern University, Evanston, IL

[†]SRI International, Menlo Park, CA

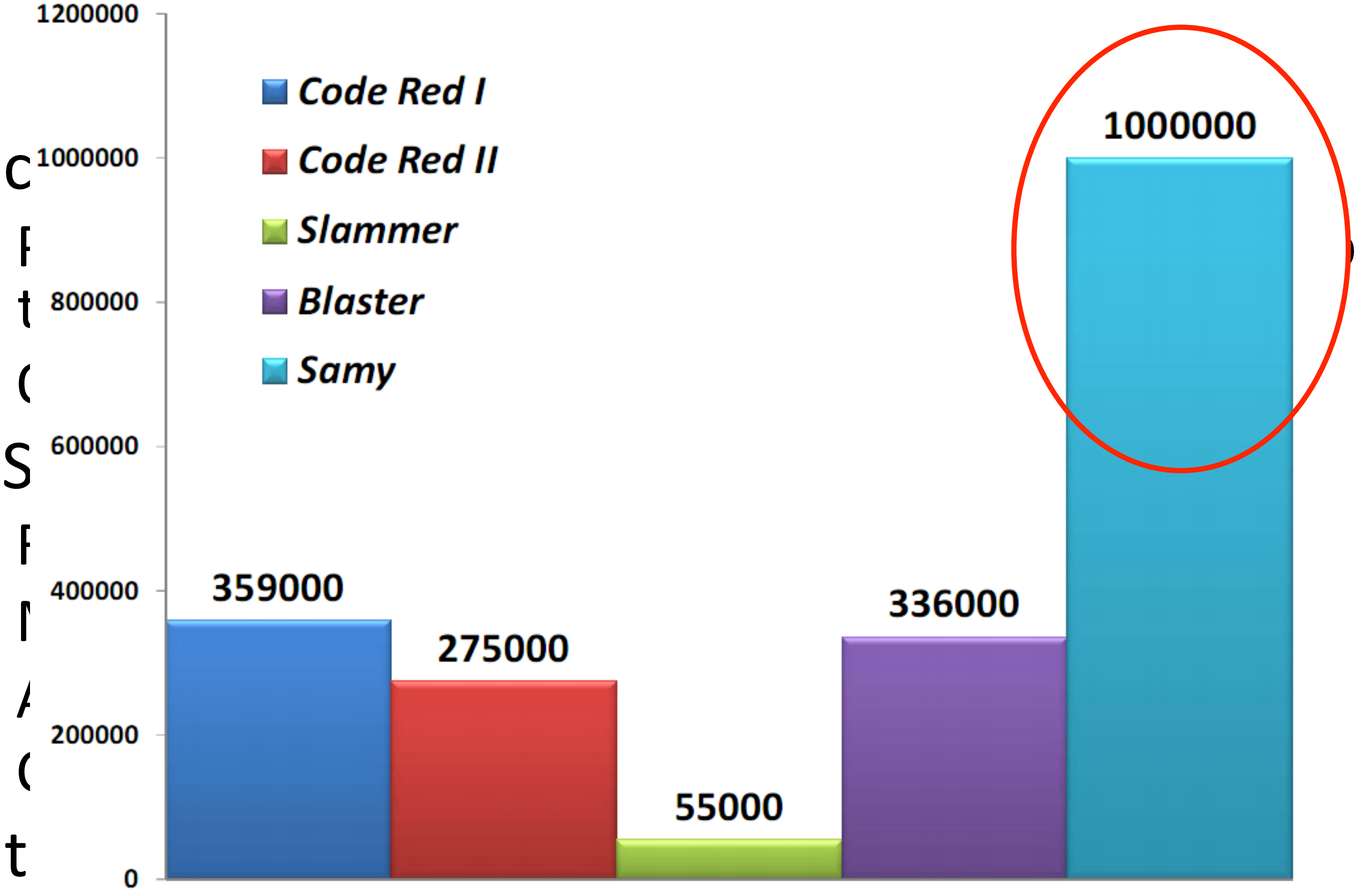

MSN Search: site:myspace.com "samy is my hero" - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://search.msn.com/results.aspx?q=site%3Amyspace.com+%22samy+
Go

[redacted]

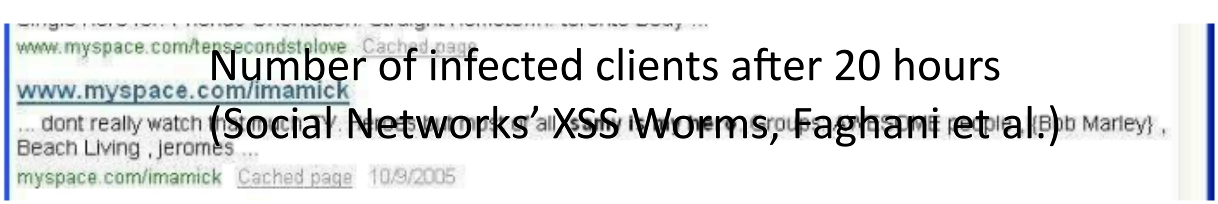
- Soc
  - F
    t
  - (
- XSS
  - F
  - M
  - A
  - (
- In t
  - Ta
  - Me

pinions,

on



Number of infected clients after 20 hours
(Social Networks' XSS Worms, Faghani et al.)



www.myspace.com/tensecondstolove Cached page

www.myspace.com/imamick

... dont really watch ... Bob Marley} , Beach Living , jeromes ...

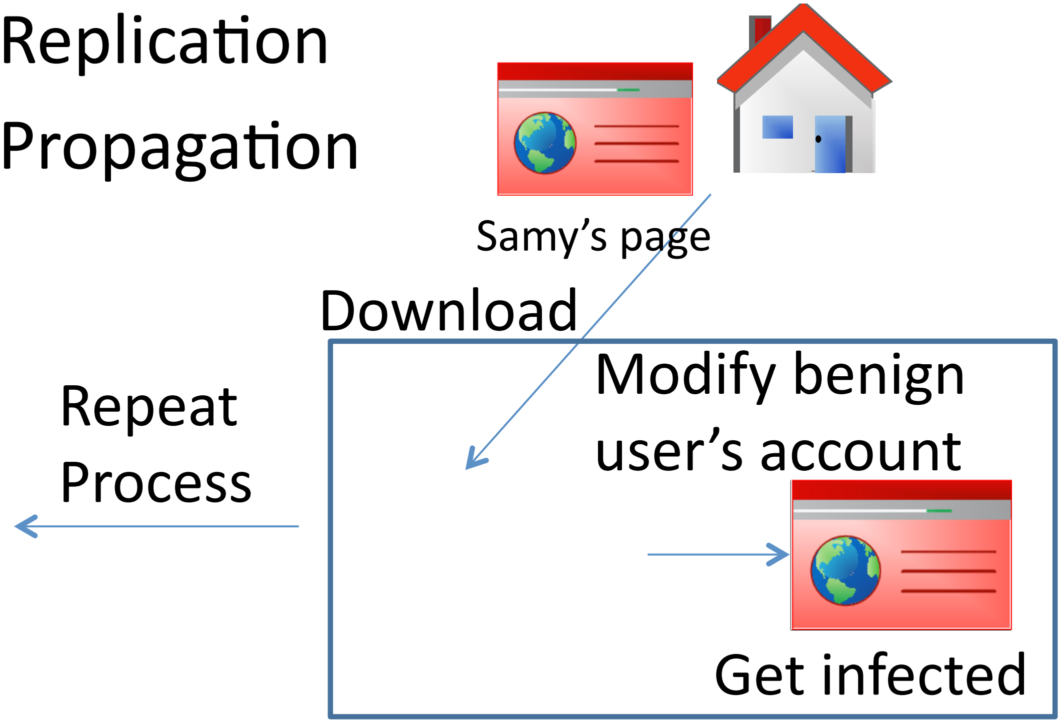
myspace.com/imamick Cached page 10/9/2005

Roadmap

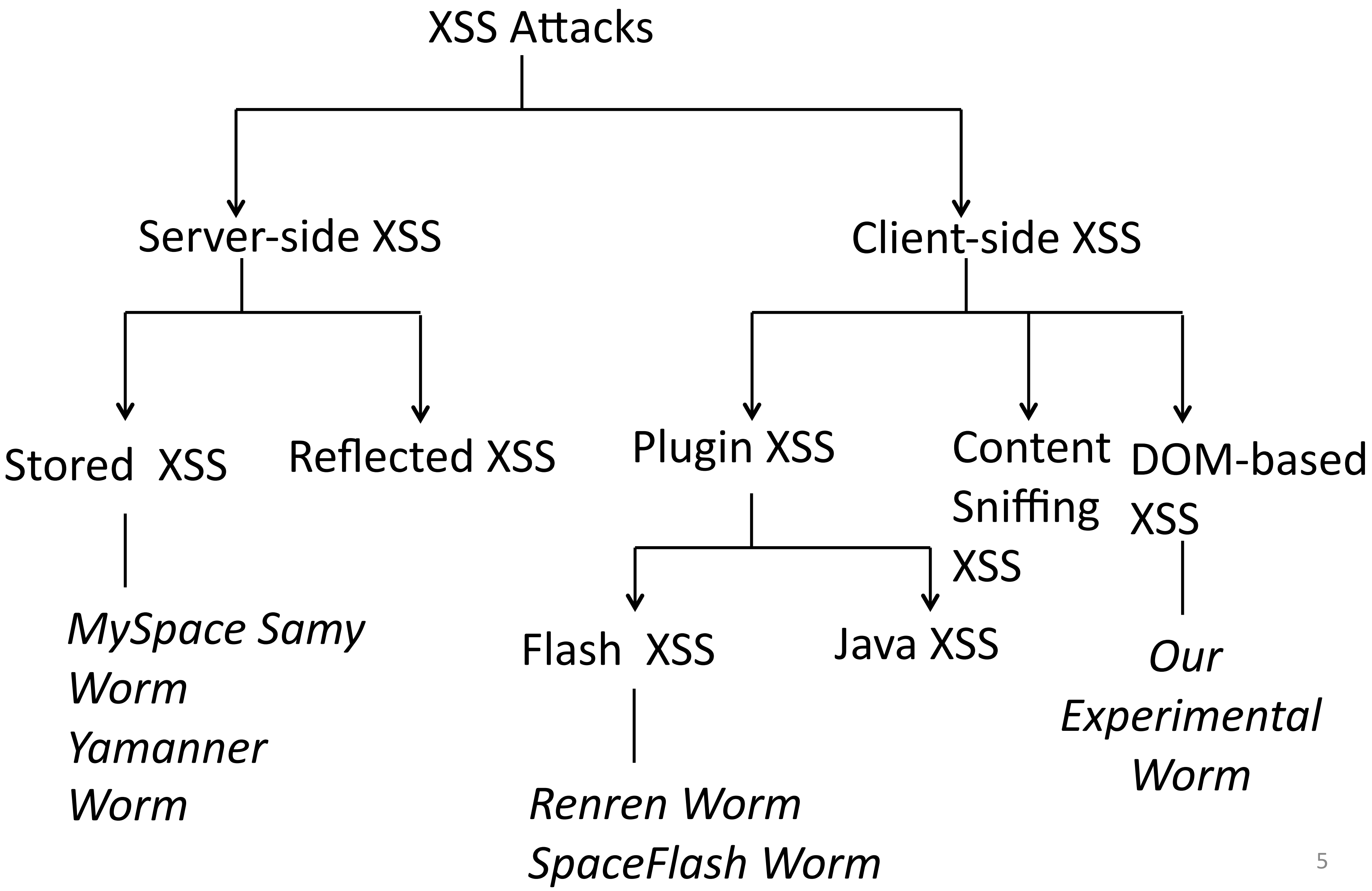
- Introduction
- Background
  - Attack Steps
  - XSS Taxonomy
- Related Work
- Our Approach
- Implementation
- Evaluation

Background

- Step 1 – Enticement and Exploitation
- Step 2 – Privilege Escalation
- Step 3 – Replication
- Step 4 – Propagation



XSS Taxonomy



XSS Attacks

Server-side XSS

Stored XSS

MySpace Samy Worm

Yamanner Worm

Reflected XSS

Client-side XSS

Plugin XSS

Flash XSS

Renren Worm

SpaceFlash Worm

Java XSS

Content Sniffing XSS

DOM-based XSS

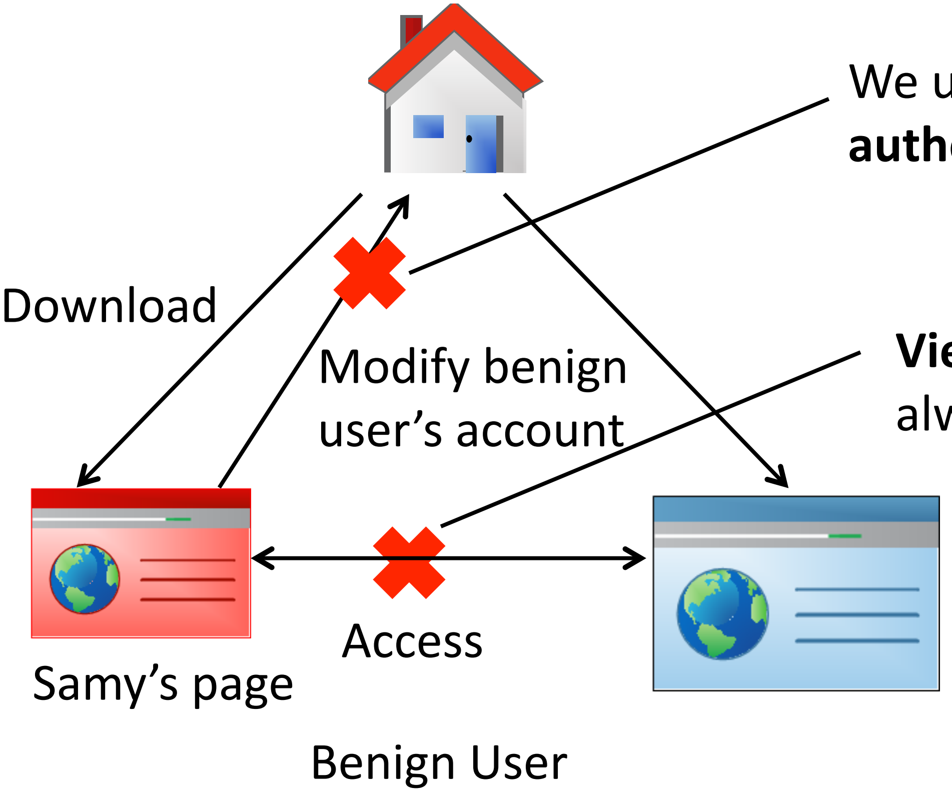
Our Experimental Worm

Related Work

- Group one: Prevent XSS vulnerabilities
  - Incomplete coverage (BluePrint, Plug-in Patches, Barth et al., and Saxena et al.)
- Group two: Prevent XSS worms
  - No early-stage prevention (Spectator and Xu et al.)
  - Not resistant to polymorphic worm (Sun et al.)
- Our goal: Prevent all the XSS worms with early-stage prevention and resistance to polymorphic worms

Our Approach

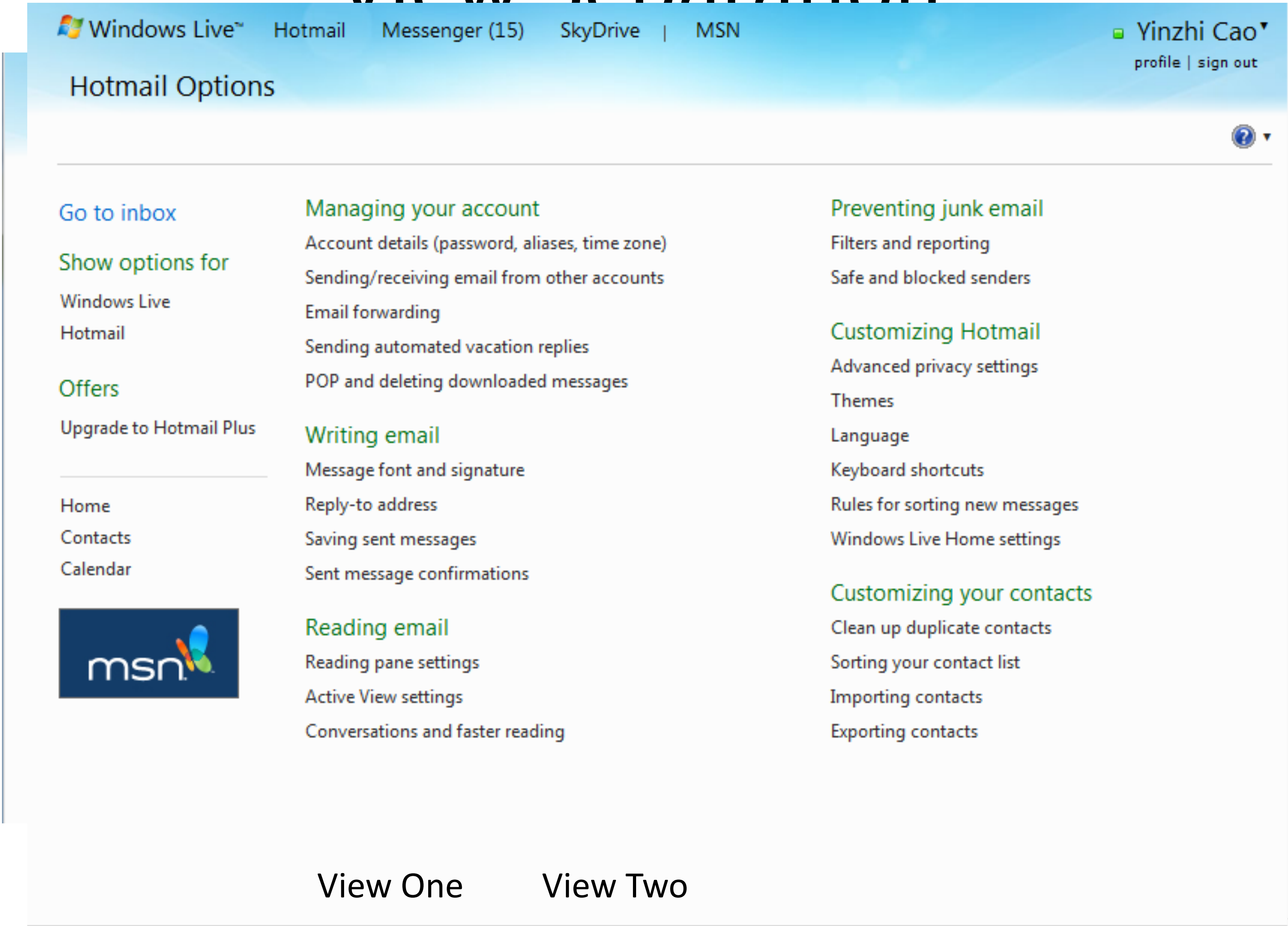
- Two key concepts: (1) request authentication and (2) view separation



We use **request authentication**.

View separation is always enforced.

View Separation



Hotmail Options

Go to inbox

Show options for

Windows Live
Hotmail

Offers

Upgrade to Hotmail Plus

Home
Contacts
Calendar

Managing your account

Account details (password, aliases, time zone)
Sending/receiving email from other accounts
Email forwarding
Sending automated vacation replies
POP and deleting downloaded messages

Writing email

Message font and signature
Reply-to address
Saving sent messages
Sent message confirmations

Reading email

Reading pane settings
Active View settings
Conversations and faster reading

Preventing junk email

Filters and reporting
Safe and blocked senders

Customizing Hotmail

Advanced privacy settings
Themes
Language
Keyboard shortcuts
Rules for sorting new messages
Windows Live Home settings

Customizing your contacts

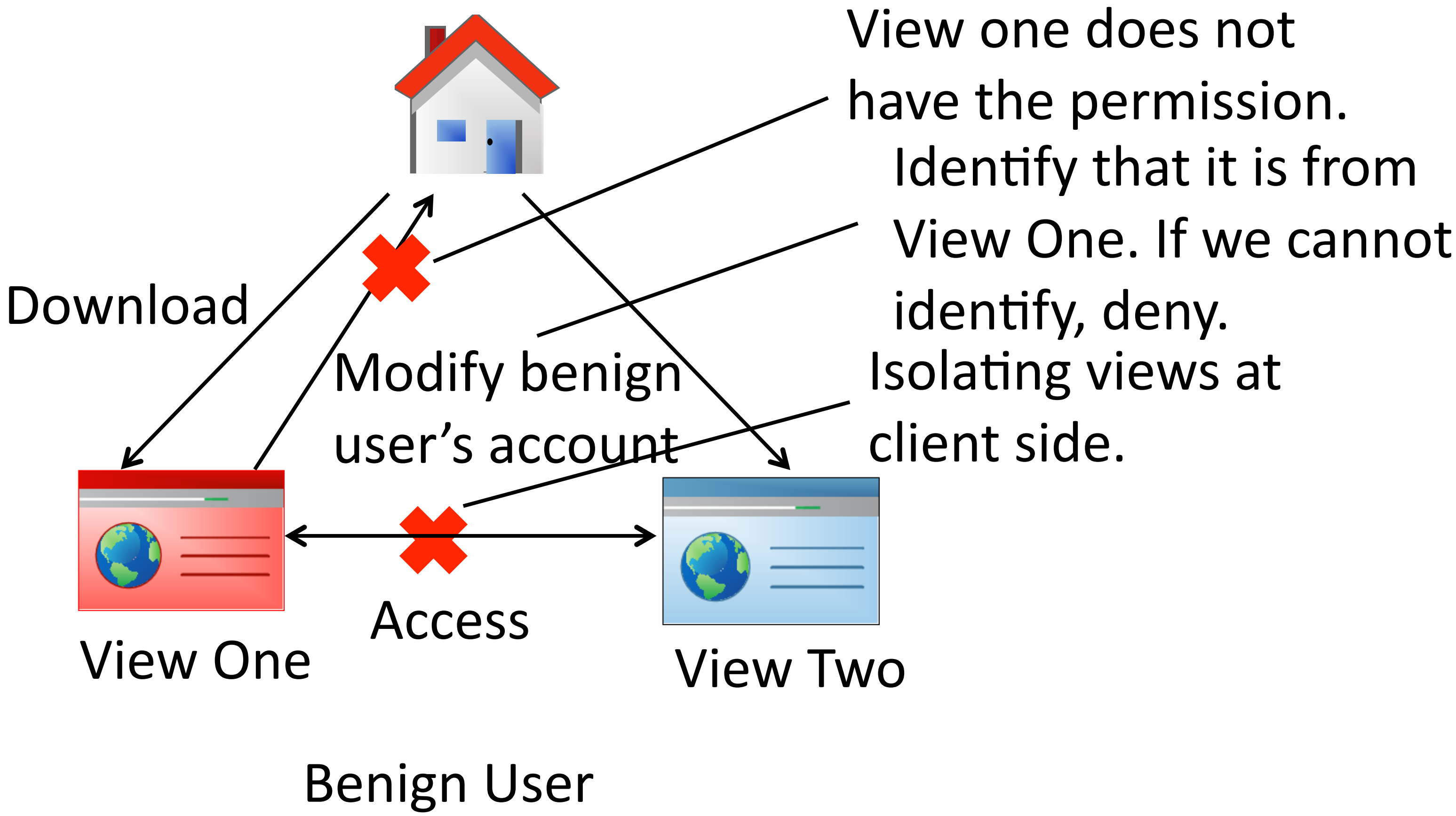
Clean up duplicate contacts
Sorting your contact list
Importing contacts
Exporting contacts

View One View Two

Request Authentication

- For example, requests from blog A does not have permissions to modify blog B
- Identifying which view a client-side request is from.
  - Secret token
  - Referer header
- Check if the view has the permission

Our Approach



View one does not have the permission.

Identify that it is from View One. If we cannot identify, deny.

Isolating views at client side.

Download

Modify benign user's account

Access

View One

View Two

Benign User

Roadmap

- Introduction
- Background
- Related Work
- Our Approach
- Implementation
  - Implementation One (Server Modification)
  - Implementation Two (Proxy)
- Evaluation
  - Case Study of Five Real-world Worms and Two Experimental Worms (only two covered in the talk)
  - Performance

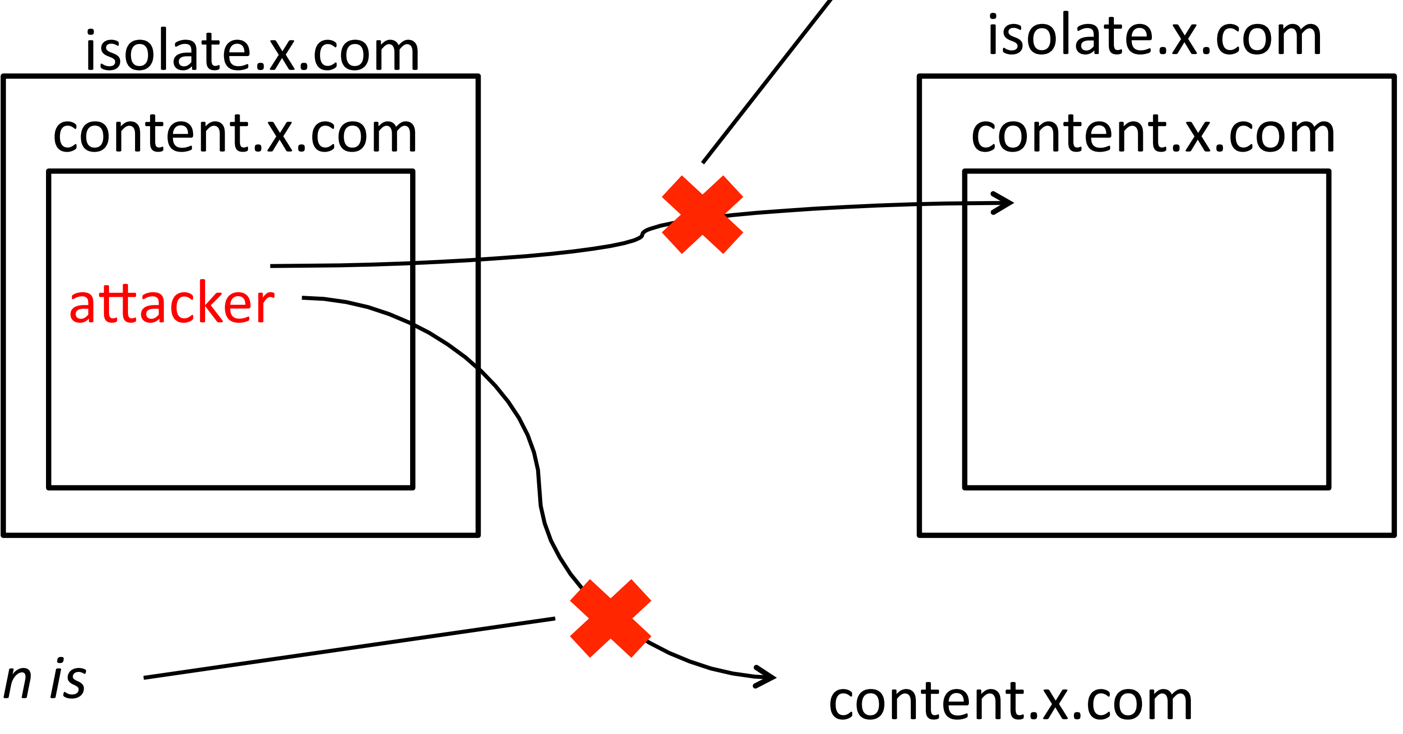
Implementation One (Server Modification)

- Prototype examples: WordPress, Elgg
- Dividing views: by blogs
- Permissions for different views: can only modify its own blog.

View Isolation for Server Modification

- Isolating views at client side.
  - – Pseudodomain encapsulation.

It cannot break isolate.x.com (different origin).



isolate.x.com

content.x.com

attacker

isolate.x.com

content.x.com

content.x.com

Secret token is required.

Request Authentication for Server Modification

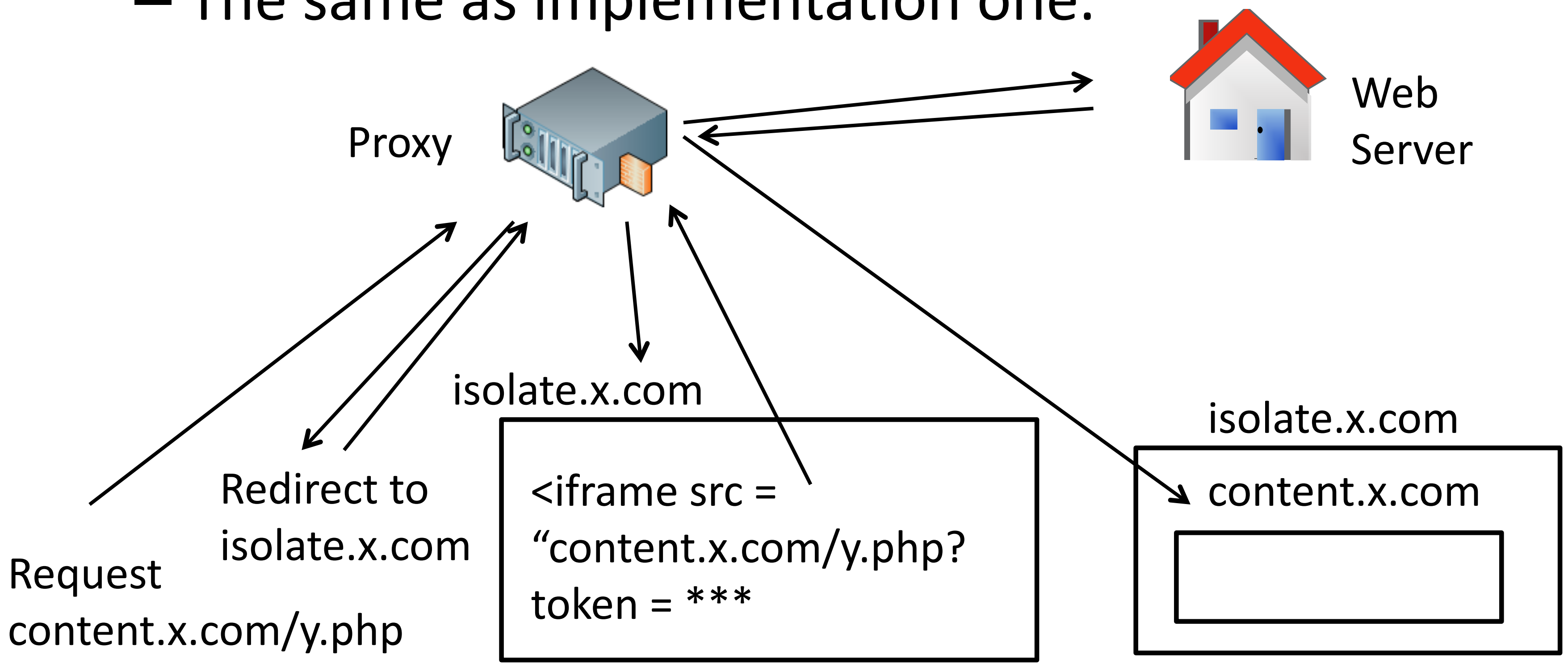
- Identifying requests from client-side
  - Secret token
    - Insertion position: Each request that will modify server-side contents.
- Checking requests' permission
  - Checking position: Database operation. (A narrow interface that each modifying request will go through.)

Implementation Two (Proxy)

- Dividing views: by different client-side URLs.
- Permissions for different views:
  - Possible outgoing post URL from those URLs

View Isolation for Proxy

- Isolating views at client side
  – The same as implementation one.



Proxy

Web Server

Request content.x.com/y.php

Redirect to isolate.x.com

isolate.x.com

<iframe src = “content.x.com/y.php? token = ***

isolate.x.com

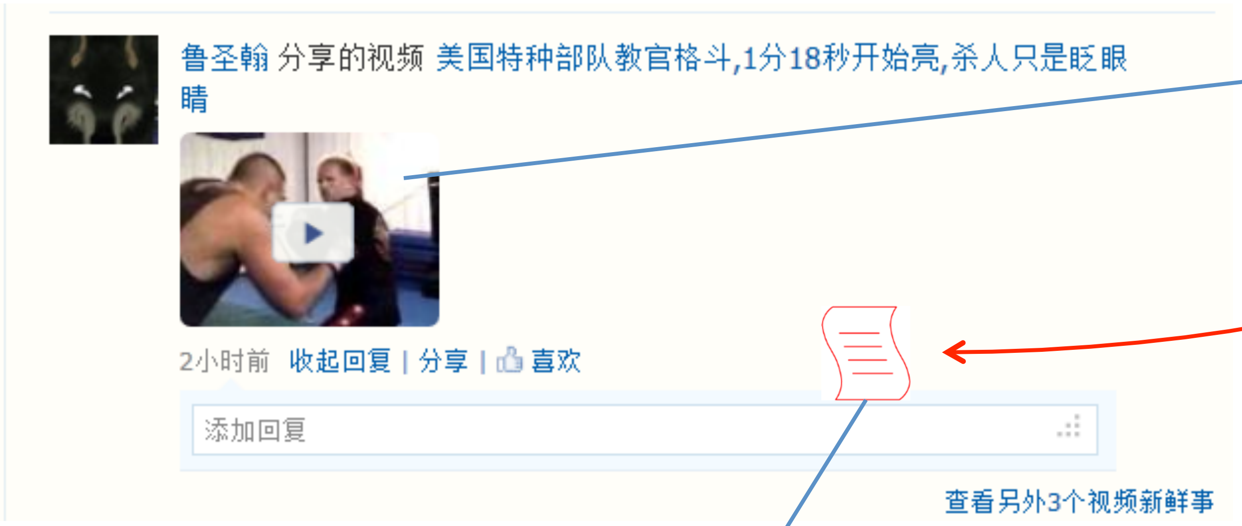
content.x.com

Request Authentication for Proxy

- Identifying requests from client-side
  - Referer header
    - Specified by the browser. Attackers cannot change it.
- Checking requests' permissions
  - Checking position: Proxy.
  - Method: See if the view has the permission to send the request.

Case Study for Real World Worms

- XSS Worm in Renren (Facebook in China)



鲁圣翰 分享的视频 美国特种部队教官格斗,1分18秒开始亮,杀人只是眨眼睛

2小时前 收起回复 | 分享 | 喜欢

添加回复

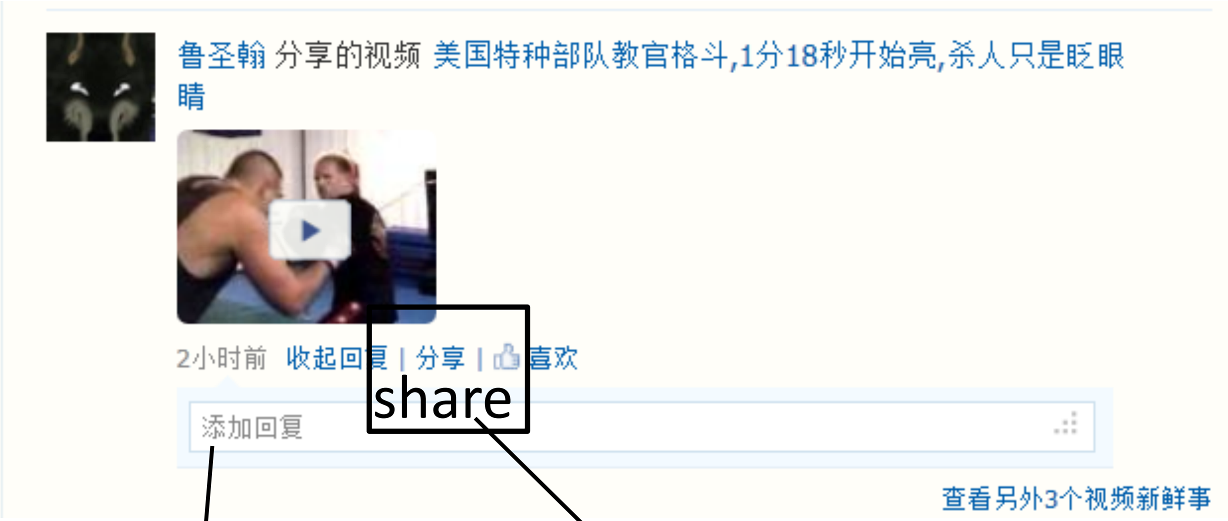
查看另外3个视频新鲜事

Flash

insert malicious scripts inside the web page

Share on the behalf of current user

View One



鲁圣翰 分享的视频 美国特种部队教官格斗,1分18秒开始亮,杀人只是眨眼睛

2小时前 收起回复 | 分享 | 喜欢

添加回复

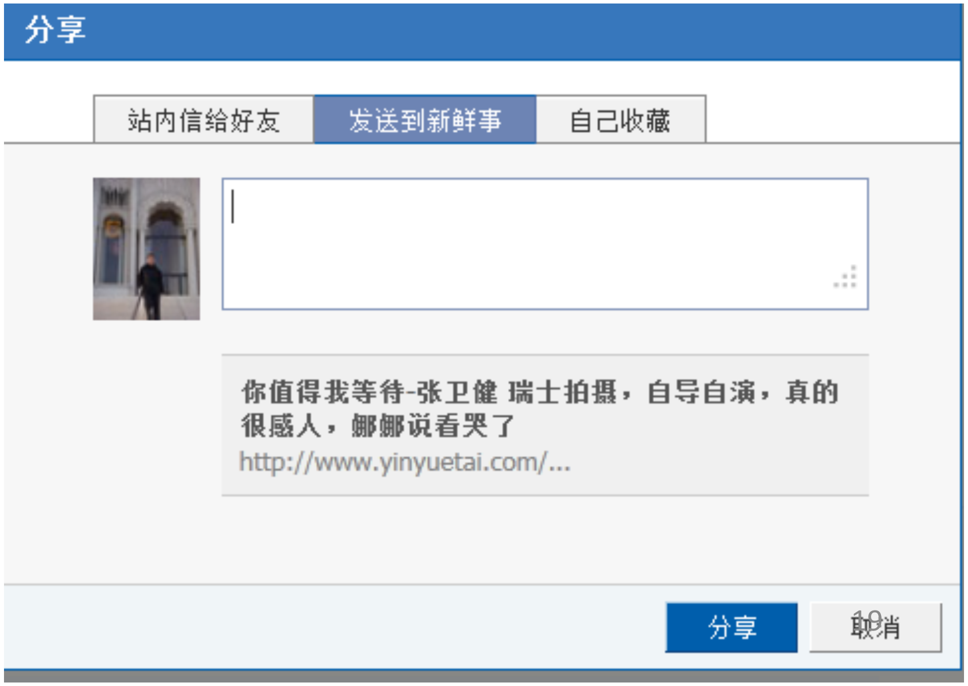
查看另外3个视频新鲜事

share

click

Request to share

View Two



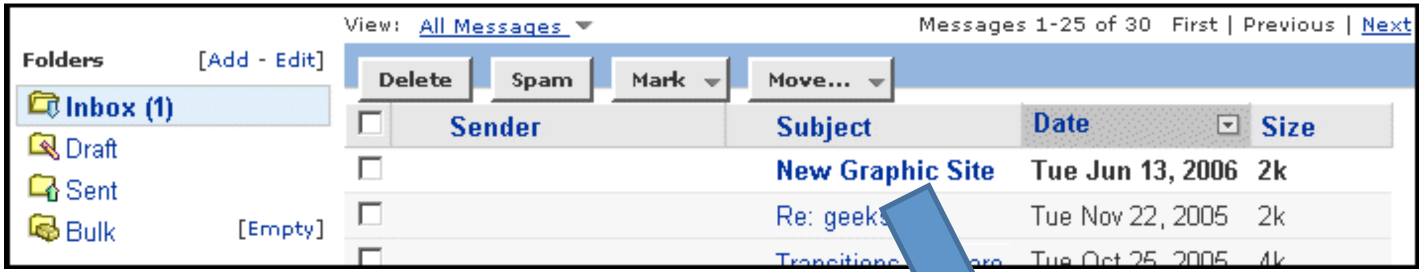
分享

站内信给好友 | 发送到新鲜事 | 自己收藏

你值得我等待-张卫健 瑞士拍摄，自导自演，真的很感人，娜娜说看哭了
http://www.yinyuetai.com/...

分享 取消

- Yamanner Worm



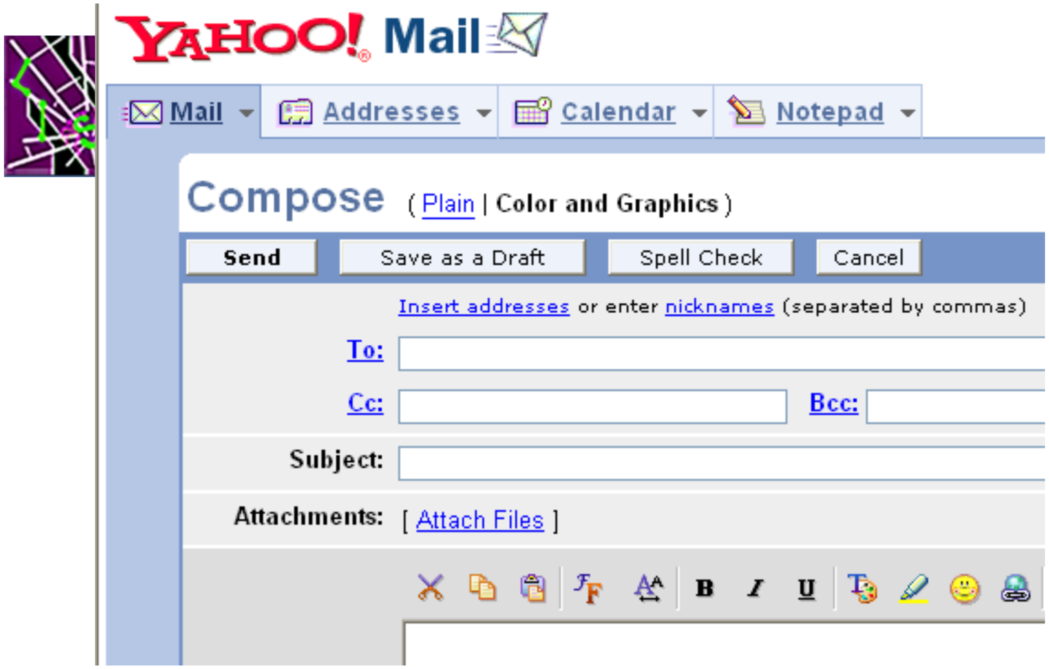
View: All Messages ▾

Messages 1-25 of 30 First | Previous | Next

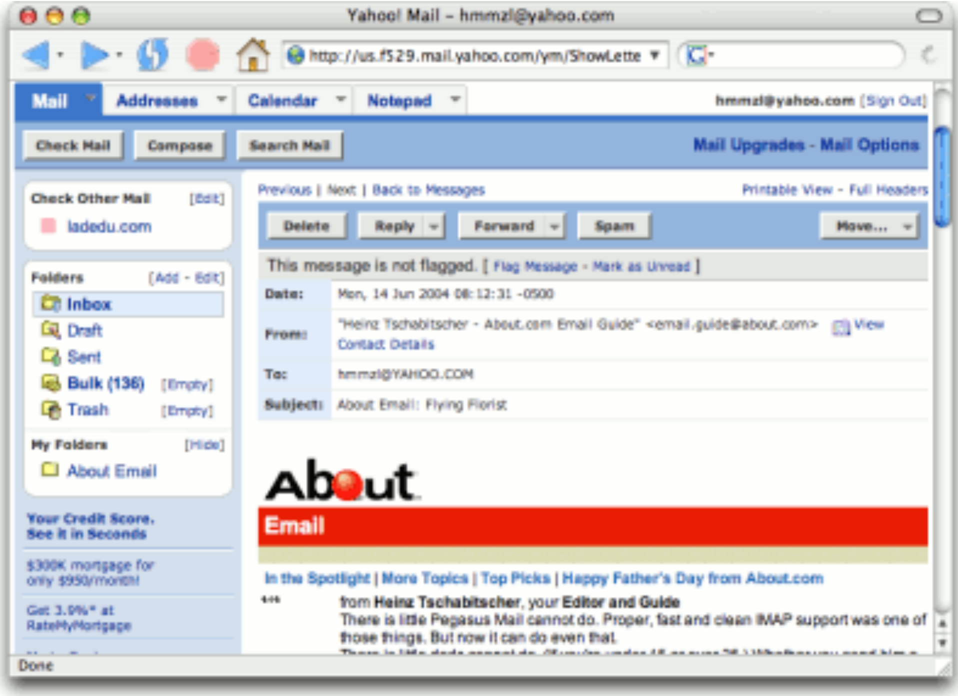
□	Sender	Subject	Date	Size
□		**New Graphic Site**	**Tue Jun 13, 2006**	**2k**
□		Re: geek	Tue Nov 22, 2005	2k
□		Transitions	Tue Oct 25, 2005	4k

Click

Send emails to
all your contacts



Insert addresses or enter nicknames (separated by commas)


Yahoo! Mail – hmmzl@yahoo.com
http://us.f529.mail.yahoo.com/ym/ShowLette
Mail
Addresses
Calendar
Notepad
hmmzl@yahoo.com [Sign Out]
Check Mail
Compose
Search Mail
Mail Upgrades - Mail Options
Check Other Mail [Edit]
ladedu.com
Folders [Add - Edit]
Inbox
Draft
Sent
Bulk (136) [Empty]
Trash [Empty]
My Folders [Hide]
About Email
Your Credit Score. See It in Seconds
$300K mortgage for only $950/month!
Get 3.9%* at RateMyMortgage
Previous | Next | Back to Messages
Printable View - Full Headers
Delete
Reply
Forward
Spam
Move...
About
Email
In the Spotlight | More Topics | Top Picks | Happy Father's Day from About.com
Done

This message is not flagged. [Flag Message - Mark as Unread]

Date: Mon, 14 Jun 2004 08:12:31 -0500

From: "Heinz Tschabitscher - About.com Email Guide" <email.guide@about.com> View Contact Details

To: hmmzl@YAHOO.COM

Subject: About Email: Flying Florist

from **Heinz Tschabitscher**, your **Editor and Guide**

There is little Pegasus Mail cannot do. Proper, fast and clean IMAP support was one of those things. But now it can do even that.

Compose email

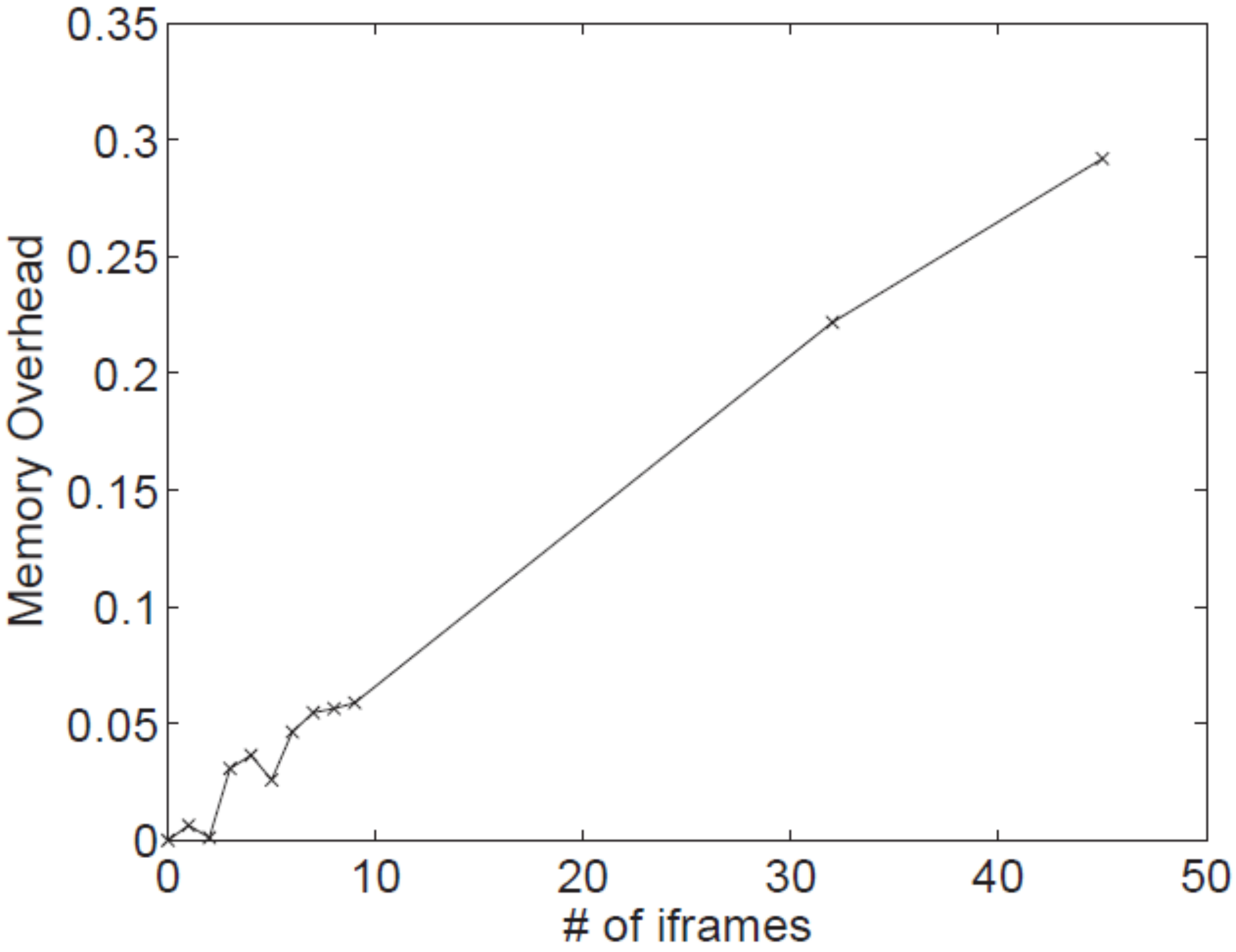
Different views

Email body

Send email ✖

Evaluation

- M
  - - ents
- R
  - -



Conclusions

- We cut off the propagation path of XSS worms through view separation by psuedodomain encapsulation and request authentication.
- We implement PathCutter by proxy assistance and server modification.
- We evaluate PathCutter on 5 real-world worms and 2 proof-of-concept worms.

Thanks!

Questions?

Backup

Comparison with Existing Works

Group Two: Worm prevention

Group One: Mitigating XSS

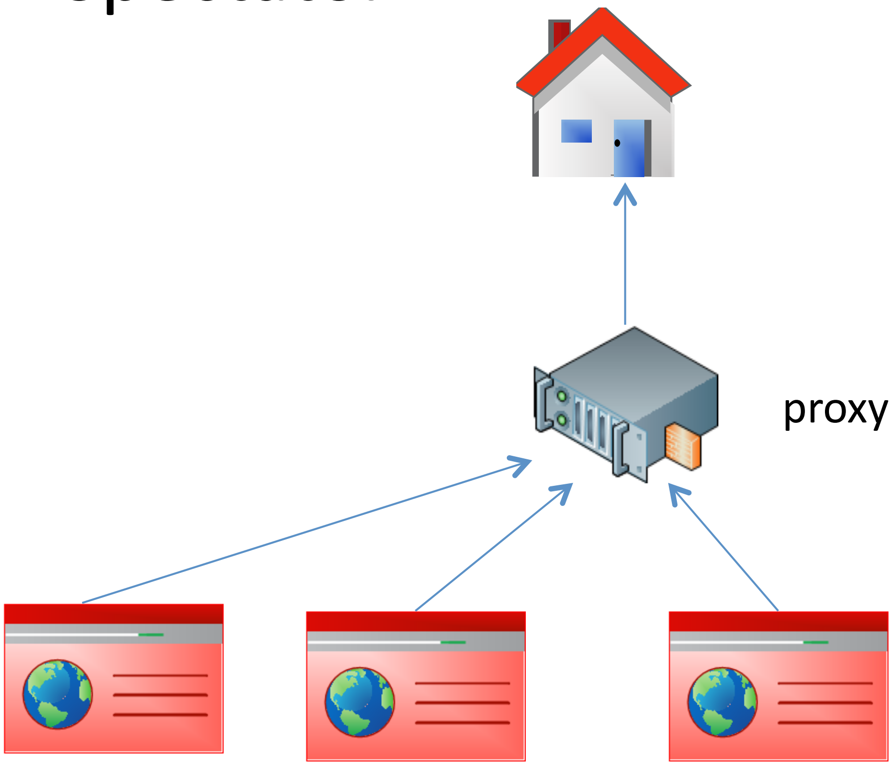
	Spectator	Sun et al.	Xu et al.	BluePrint	Plug-in Patches	Barth et al.	Saxena et al.	PathCutter
Blocking Step	4	3	4	1	1	1	1	2
Polymorphic Worm	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes
Early-Stage Prevention	No	Yes	No	Yes	Yes	Yes	Yes	Yes
Types of XSS that Can Be Defended	All	All	Passively Observable Worms	Traditional Server-side XSS Worms	Plug-in XSS Worms	Content Sniffing XSS Worms	DOM-Based XSS Worms	All
Deployment	Server or Proxy	Client	Server	Server	Client	Client	Client	Server or Proxy
Passive/Active Monitoring	Active	Passive	Passive	Active	Active	Active	Active	Active

Limitation

- Need to know the semantics of web application
- Only prevent worm behavior but not all the damages

Existing solutions

- Spectator



proxy

the same injected tag

the same injected tag

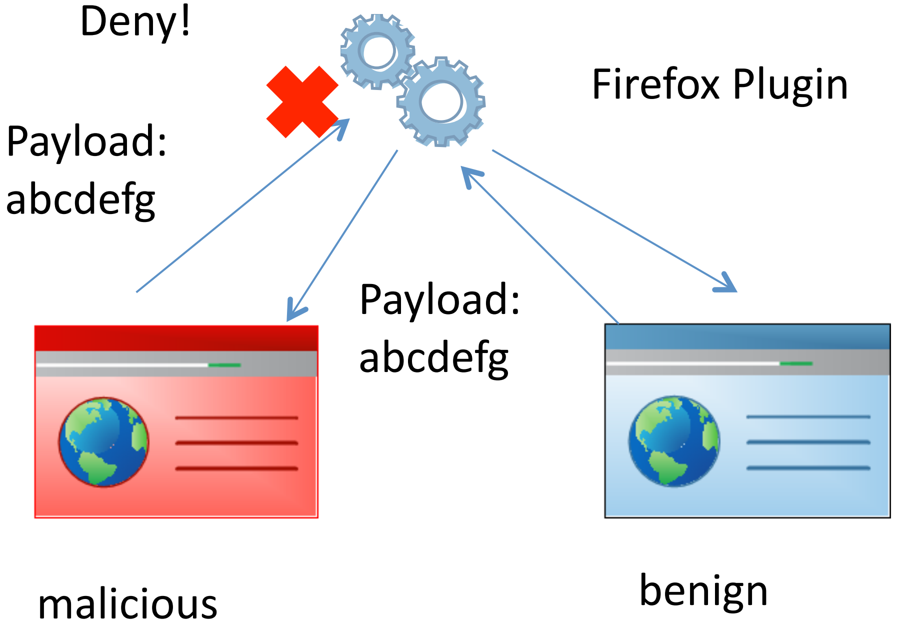
But it can only detect the worm when it spreads for a while!

… if it reaches a threshold, report it.

Existing solutions

- Esorics 09

But
(1) Payload may change.
(2) Pure client-side solution.



The same payload
Deny!

Firefox Plugin

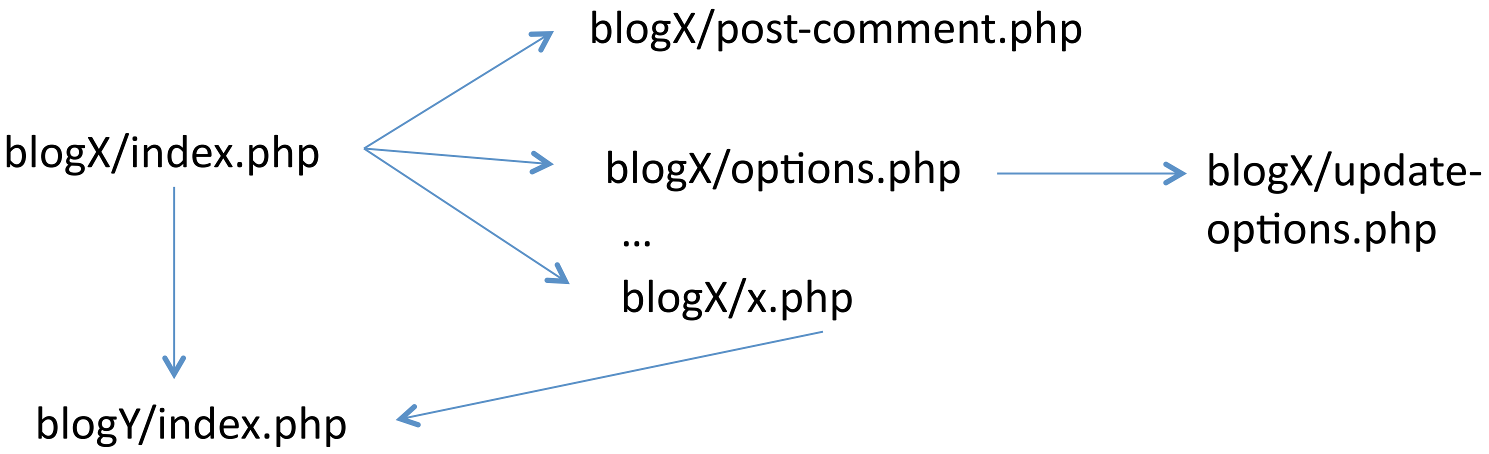
Payload:
abcdefg

Payload:
abcdefg

malicious

benign

URL graph provided by the server or a third-party



blogX/post-comment.php

blogX/index.php

blogX/options.php

blogX/update-options.php

…

blogX/x.php

blogY/index.php