Whistling over the wire

Arnauld Mascret

Plan

1 Presentation
- Subject
- Previous work
- Today subject

2 Architecture

3 Planning the attack

4 Scenario

5 Conclusion

Quick survey

- Member of Sogeti ESEC R&D team
  - Focus on information gathering from open sources (OSINT)
- Our red team job (try to get in, "without pre-defined perimeter")
  - The Intel phase (my part)
    - Find the target
    - Learn about it
  - Define a scenario of attack
    - Find a weakness/vulnerability
    - Make/adapt an exploit (not in this presentation)
    - Sometimes, trick the target to run the exploit
  - Get in, take the data, then disapear

Preparing targeted attack

Needed information

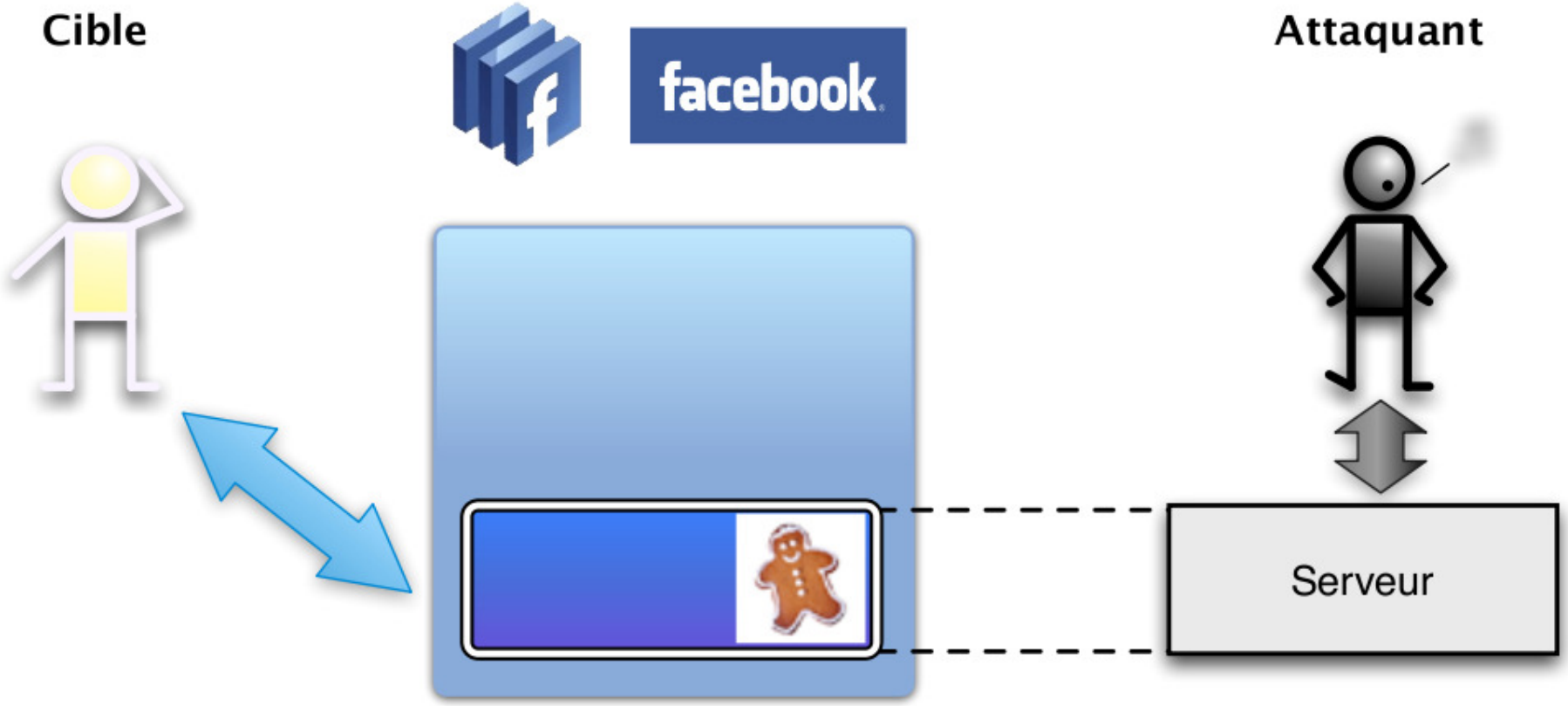
- One or more human targets
  - With access to the data
- Knowledge about the target computer
  - Operating system
  - Software and version
  - Protection
- Trusted connection
  - Method to usurpate identity
  - Enough knowledge to make it look real

The “easy” way

HITB Dubaï 2010

- Find the target : LinkedIn
- Build trust : identity theft on Facebook
- Gather Knowledge : Facebook Application and Javascript
- How to target : Facebook Id plus specific Application code

The “easy” way



Today subject : Twitter and microURL

Twitter

- Users follow other users
- Share short message, sometimes with link to other ressources

microURL

- Share link with shortcut URL

Twitter and microURL

microURL

- Usefull in Twitter, message do a maximum of 140 char
- Service using redirection for shortened link in messages
- Exemple source :
  - `http://www.youtube.com/watch?v=oHg5SJYRHA0`
- Exemple of shortened URL :
  - Shortened Link : `http://tinyurl.com/2g9mqh`
  - Shortened Link : `http://ow.ly/7IyQ6`
  - Shortened Link : `http://to.ly/TR`

microURL

- Hide the real destination
- **What if the micro Url service is malicious ?**

Common knowledge

Twitter

- Javascript Worm using XSS
- Application Scam
- Stolen account (Barack Obama, Britney Spears)

MicroURL

- SPAM, Virus, redirection to browser exploit toolkit
- w2spconf 2011 (Alexander Neumann, Johannes Barnickel, Ulrike Meyer)
  - List and explain many risks

Why a new presentation

Previous works focus on vulnerabilities and large scale attack.

My question was : Is it possible to use the combination of Twitter and micro url to make a **stealth targeted** attack ?

Starting point :

- I have a name and a Twitter account

Objectives :

- Can I use this to get access to the computer ?
- And how ...

Plan

1 Presentation

2 Architecture
- Twitter
- Url Shortening Services (USS)

3 Planning the attack

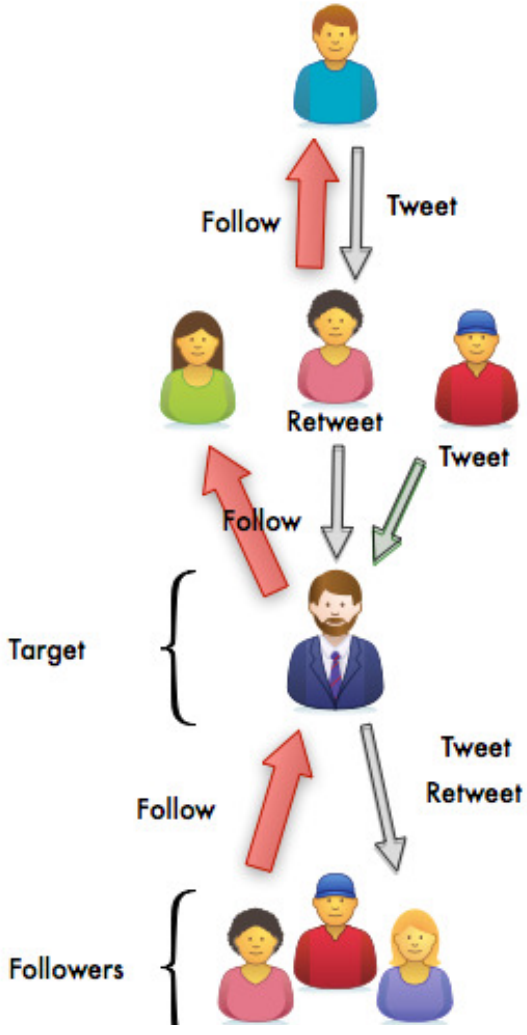
4 Scenario

5 Conclusion

Twitter

Definition

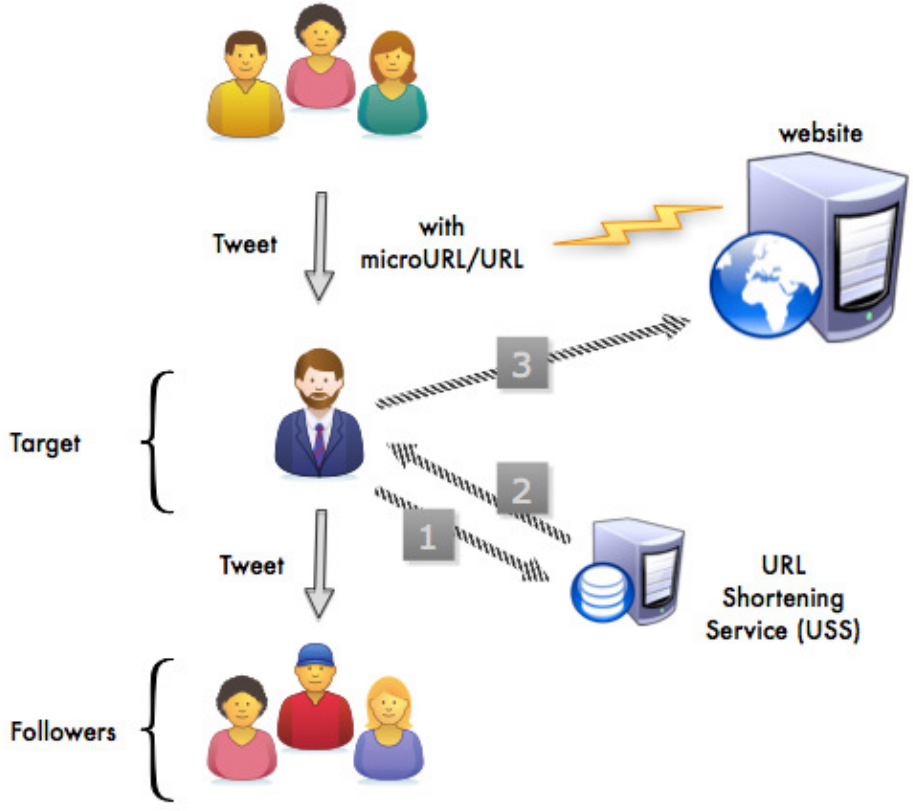
- Tweet: short message
- Following a user : In order to get a notification when user posts a new message
- Follower: person following someone
- Hashtag: add keyword using # (#hitb)



Twitter & USS usage

Information flow

1. A tweet is sent with a microURL to a website
2. Target uses the microURL to get the final URL
3. The service sends back the final URL (or an other microURL)
4. The user accesses the website



How it works ?

These are services that redirect a short URL to a long URL.

Methods

- HTTP redirection with 30X status code
  - 301 : Moved Permanently
  - 302 : Moved Temporarily
- META Header
  - `META http-equiv="refresh" content="0;URL="`
- Javascript
  - `location.replace()`
- Link

Some results

A lot of services exist (more than 300 in 2009)

On 31 USS tested

- 19 : 301 redirection
- 9 : 302 redirection
- 1 : meta and javascript (t.co)
- 2 : links with previsualisation

Results depend of the user agent (t.co)

One known and legitimate service use javascript

Starting strategy

First step : promote my rogue Url Shortening Service

- Create twitter use
- Post shortened link using it

Second step : make my target use it

- Send him a link
- Even better if someone he is following sends him the link

Third step : learn about the target

- When he uses our service

Final step : Exploit the target

- When he uses our service a second time

Plan

1 Presentation

2 Architecture

3 Planning the attack
- Objectives
- Twitter
- Attack surface

4 Scenario

5 Conclusion

Information needed

Known connection to impersonate

- User
  - Language
  - Style
  - Media
- Website
  - Subscription (miles card)
  - Facebook (application)
  - News

Technical Information

- System
  - OS
  - Software and versions
  - Browser and plugin
- Defensive mechanisms (e.g. AV, FW, ...)

Twitter: Gathering Information

Method

- Parsing HTML
  - twitter.com
  - Ajax
  - mobile.twitter.com
- API
  - Double authentication (user, application)

Twitter and spammer

Twitter filed a suit in federal court in San Francisco against five of the most aggressive tool providers and spammers (5 April 2012)

Twitter API

User

- Date of creation : "Wed Jun 24 22:24:43 +0000 2009"
- Location : "San Francisco"
- Followers
- friends_count
- number of tweets

Tweet

- Date of creation
- Mentionned users (@)
- Urls, hashtag (#)
- Reply to
- Geolocalisation
- Source: TweetDeck

Micro URL: Information available

Serveur web : redir 30X

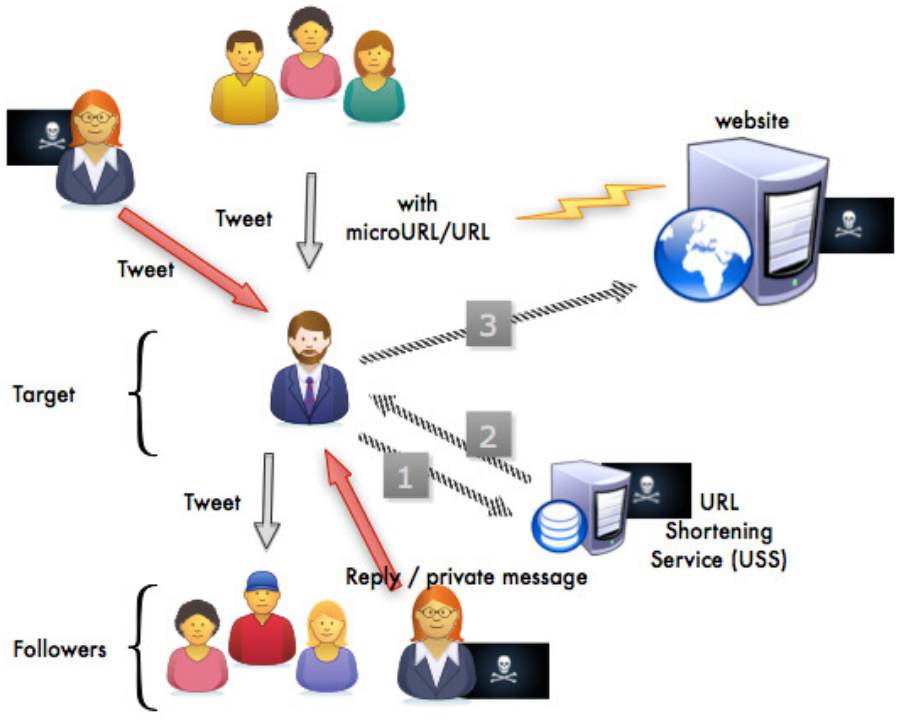
- User Agent
- Referer : t.co
- Date, time
- IP, country

Javascript

- Browser, version
- Plugin

Attack surface

- Send link to our USS
  - Tweets
  - Reply
  - Private message
- Interception
  - URL Shortening service (rogue, hacked)
- Destination could also be a solution
  - Website (rogue, hacked)



Scenario updated

- Create many Twitter users
  - Same hobbies (#)
  - Similar thematic for tweet and retweet
  - Follow same sources (user)
- Send link
  - Tweet, Reply, Direct message
  - With a link using own Url Shortening Service
- Identify when the target uses our service
  - Gather information to find a vulnerability
  - Get control now or on second shot

Plan

1 Presentation

2 Architecture

3 Planning the attack

4 Scenario
- Account creation
- Making relation
- How to get follower
- Identifying the user
- DEMO

5 Conclusion

Automatized account creation

Requirements

- Full name : no constraint
- Twitter name : unique
- Email to validate your account : unique

The email bottleneck

Email

- Temporary account
  - google : xxxxx-yyyy@gmail.com
- Reuse the same gmail account : xxxyy.yy@gmail.com
  - google : xxx.yy.yy@gmail.com
  - google : xxx.yy.y.y@gmail.com
  - google : xxx.y.y.y.y@gmail.com
- Temporary account
  - anything@yopmail.com
  - With easy-to-parse RSS : www.yopmail.com/en/rss.php?login=ann_onym

Finding available email and id

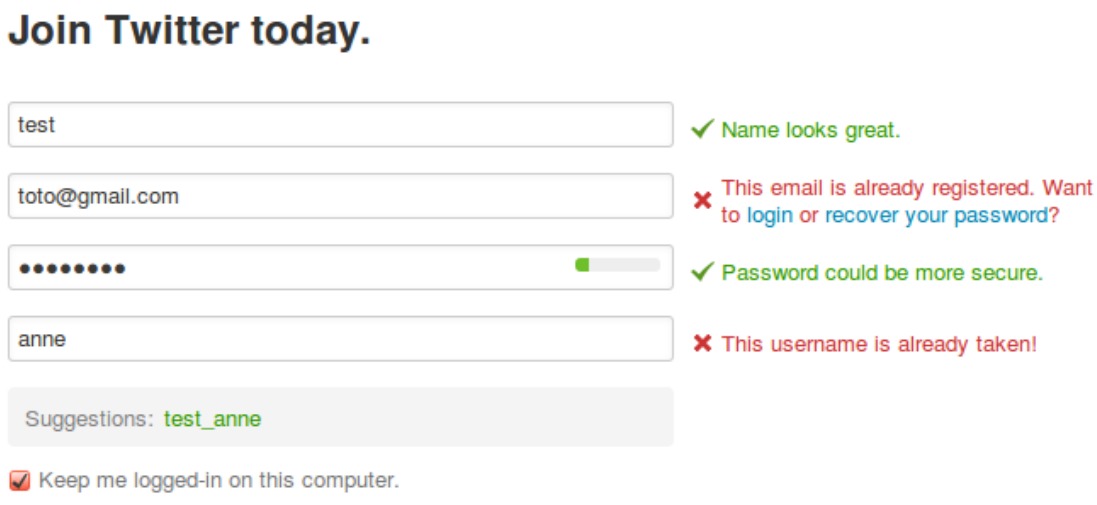
Twitter uses a lot of ajax calls to make it user-friendly

ID

```
https://twitter.com/users/username_available?username=ann
```

Email

```
https://twitter.com/users/email_available?email=a@gmail.com
```



Captcha

Twitter.com

mobile.Twitter.com

Captcha Mobile

Vulnerability

- Id of captcha in the form
- Possible to retry if fails
- Can't replay if is correct

OCR

- Always 6 integers
- Fixed police, font
- No real deformation, but anti-aliasing
- Few noise
  - Always with some kind of blue
  - Straight line

Adding relations

Web based

- Normal (AJAX)
- Mobile (POST)

Using application

- User identification with Oauth
- Application identification
- Visual picture displayed to user for validation

The first followers

The bot method

- Make bots and cross add
- Or buy them : 5$ to 15$ for 1000 followers

The follow back

- Get a list of followers often following back
- Add lots of them

The hard way

- Make it real and interesting
  - Exemple : Lady Gaga
- Or tweet about iphone jailbreak
  - Exemple : 0naj, i0n1c, pod2g

Identifying the target

The hardest part

Twitter	**USS**
geolocalisation and country	IP
url in tweet	redirection and cookie
date of tweet, retweet	date of redirection
language	language in header
source (tools) and os	user agent, js detection

Filtering the potential user

Target : @ivanlef0u

- url : ivanlef0u.fr
- timezone : paris
- hashtag : security, microsoft
- use android, chrome and silver bird

Problem

That is a lot of potential target. We want to filter as much as possible to stay stealthy.

Getting out of the box

With Facebook

- Set a specific cookie with an application for each user
- Check the cookie with your USS
- Retrieve the facebook user id

History leak

- Use Icamtuf solution
- Compare with the list of urls in tweet

A cool trick that doesn't work

- Check for specific url with user ID
  - Twitter : twitter.com/user/lists
  - Facebook : profile.phpid=xxx&sk=approve

History extraction

source : `http://lcamtuf.coredump.cx`

Solution

- Create iframe with url to test
- Constantly access the content of iframe before X seconds
- If cross-domain policies access error then page in cache
- Else : change the destination url to prevent cache damage

Performance

- Works with css, picture, js, ...
- 50 urls per second
- Using cookie allows to dispatch test in time

Filtering the potential user

Target : Ivan Le Fou

- His blog is a wordpress
- He should be the only admin
- There is an admin section with picture, so we have distinctive urls

Configuration of the micro Url Service

1. A first redirection : 1 in javascript to point 3 and 1 slower with meta to point 2
2. Just go to the destination, JS is not active
3. If ip is french, and time of access is in the afternoon, and userAgent is about windows, then set a specific cookie for next time
4. If a cookie is set, load JS to check for URL

Scenario with IvanLeFou

Target : VM windows 7

- Browser with old silverlight plugin
- Use link in tweet

Attacker's VM

- The Url Shortening Service
  - If specific URL found redirect to pwn.html then to the final destination
  - If not redirect to final destination

Scenario

- Test without history
- Test with history set (a picture on the admin side of his wordpress blog)

Demo

DEMO

Plan

1 Presentation

2 Architecture

3 Planning the attack

4 Scenario

5 Conclusion

Conclusion

Traditionnal SE

- Often direct
  - Mail with document
  - Mail with URL
- Sometimes targeted
- Users tend to learn

Using twitter

- Less direct if retweet
- Harder to know who was the target
- User seems more prone to click on links in tweets

Conclusion

Risk is low

- Does not break the internet
- Using different clients make the information correlation harder
- Javascript is a big tell
- Need other vulnerabilities (0day)
- Twitter make spam difficult

But it does work

- The architecture of Twitter + microUrl seems weak
- All the tools already exist

Thank you for your attention
Any questions?

Laboratoire **Sogeti-ESEC**
24 rue du Gouverneur Général Eboué
92136 ISSY LES MOULINEAUX - France

arnauld.mascret@sogeti.com

SOGETI