Taming the B.E.A.S.T.

Browser Exploit Against SSL / TLS

by Richard Newman, CISSP
GIAC - GPEN, GIAC - GWAPT

about Richard Newman

- Senior Information Security Engineer with a large local retail headquartered in central Florida
- 23 years of diverse IT experience with the last 8 years focused within Information Security
- Specializing on network security, vulnerability analysis, network and application penetration testing, and computer forensics

Index

- What is the B.E.A.S.T.?
- What is not affected
- Requirements
- Sample attack scenario
- Mitigations
- Detail look at how the attack works

- References

What is B.E.A.S.T.

Brute force attack against a browser's session where plain text of the encrypted communication can be obtained

Possible due to the chosen encryption mechanism Cypher Block Chaining (CBC) and the flaw discovered by W. Dai

Attack expanded by Rizzo and Duong

But: Only in certain circumstances

What is not affected

- HTTPS using TLS 1.1 or higher
- Protocols other than HTTPS
  - Imap
  - POP
  - SMTP
  - FTPS
  - SSH
- Post Data
- HTTPS using compression (not found very often)

Requirements for B.E.A.S.T to work

- Javascript enabled in browser
- Encryption using SSL v3 or TLS v1.0
- Able to packet capture communications
- Able to modify packets sent from you
- Browsing with multiple tabs/sessions
- Attacker must have an idea where you are going to browse
- Attacker must be able to perform their action(s) within the time you are logged in

Sample Attack Scenario

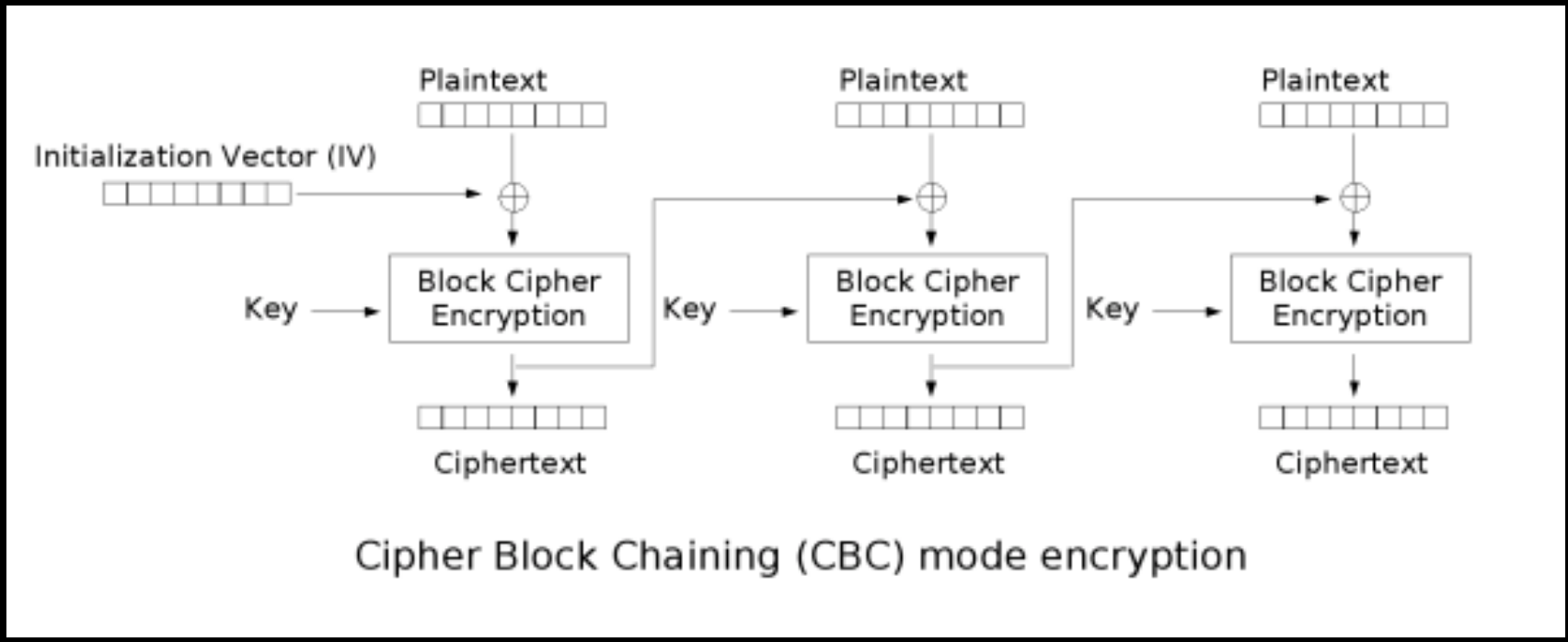
- Start Browser session
- Have javascript enabled in browser
- Browse to <evil> site
- Open new tab and browse to secure site (bank, paypal, etc)
- Keep browsing for 10 min

- The attacker now can use the session cookie in this example to login as you (but only as long as you are still logged in)

Mitigations against B.E.A.S.T.

- Close your browser session between site visits
- Disable javascript
- Do not browse to both insecure (http) and secure (https) websites within the same browser
- Do not visit unknown sites

Cipher Block Chaining



Cipher Block Chaining (CBC) mode encryption

Attack Detail

Remember we can sniff and alter packets

Consider a conversation between Alice and Bob

- You observe a record which contains Alice's password in block i - M(i)
- You have a guess of the password P
- You know that the next block will be encrypted with Initialization Vector X and you can inject a block
  - You inject X xor C(i-1) xor P
  - when this gets encrypted X get Xored in (negate the IV this makes the injected packet look like the previous packet)
  - C(i-1) xor P gets fed to the encryption algorithm
  - if your encrypted guess Ci equals Alice's encrypted password Mi you have guessed correctly!!!

Attack Detail

- Refinements to original Dai attack
  - CBC is block oriented
  - HTTP protocol allows for padding
    - <!-- can you say comments -->
  - We can use this to shift the data we want to reveal
  - Or add a piece of data we want to check / discover

References

- http://luxsci.com/blog/is-ssltls-really-broken-by-the-beast-attack-what-is-the-real-story-what-should-i-do.html
- http://www.theregister.co.uk/2011/09/19/beast_exploits_paypal_ssl/
- http://netifera.com/research/beast/
- http://blog.tempest.com.br/static/attachments/marco-carnut/driblando-ataque-beast-com-pasme-rc4/ssl_jun21.pdf
- https://blog.torproject.org/blog/tor-and-beast-ssl-attack
- http://www.educatedguesswork.org/2011/09/security_impact_of_the_rizzodu.html
- http://en.wikipedia.org/wiki/Block_cipher_modes_of_operation