
Tor in the Web2.0: understanding and avoiding the new privacy pitfalls



Rilasciato sotto Creative Commons Attribution
Share-Alike3.0 Unported
<http://creativecommons.org/licenses/by-sa/3.0/>



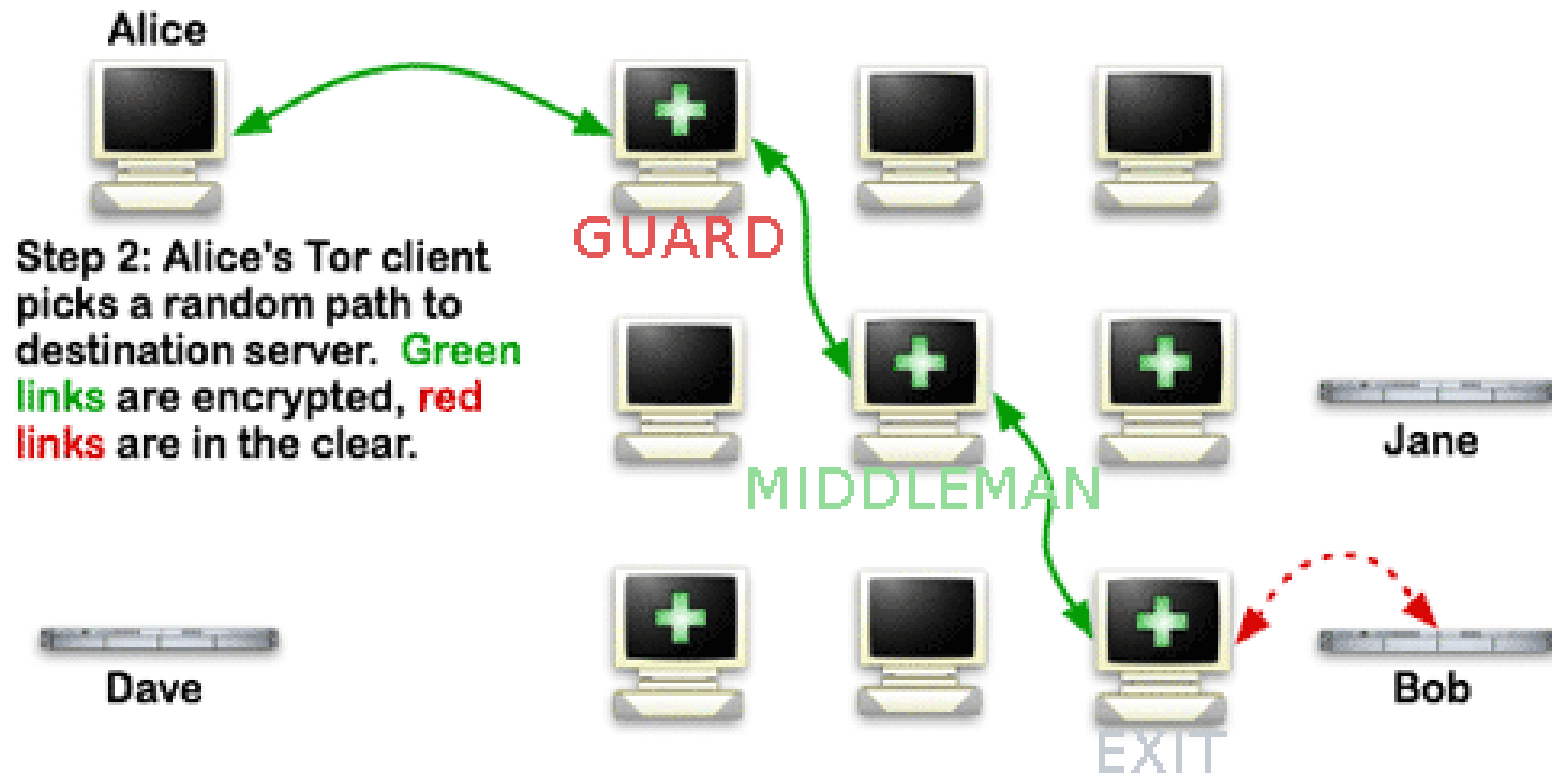
Scaletta

- Tor Crash Course
- Web 2.0
 - HTML 5
 - Client Side Storage
 - Custom Protocol Handlers
 - Offline Web Applications
 - Multimedia
 - Browser Geolocation
 - Eye Candy
- Conclusioni



Tor Crash Course

How Tor Works: 2



Client Side Storage

- Ottimo lavoro eseguito da Alberto Trivero sull'argomento
- Cosa offre?
 - Session Storage
 - Local Storage
 - Database Storage



Session Storage

- Sono dei cookie *on steroids*
- Legati al dominio di creazione
- Legati alla finestra correntemente visualizzata
- Distrutti con la chiusura della finestra



Local Storage, Database Storage

- Associato al dominio correntemente visualizzato
- Accessibile da qualsiasi finestra
- Distrutti dall'applicazione che li utilizza, persistono alla chiusura della finestra
- Associato esclusivamente al dominio di creazione
- Un completo database relazionale lato client
- Controllato dall'applicazione che lo utilizza



Client Side Storage

- Supponiamo che, di per se, la tecnologia sia sicura (AHAHAH!, cfr. Trivero)
- L'iniezione di codice da parte di nodi malevoli diventa un fattore decisivo
 - Inserimento codice di estrazione dei dati
 - Inserimento codice di trasmissione dei dati verso macchine controllate dall'attaccante



Client Side Storage

- Il database storage non è ancora molto diffuso
- Per (s)fortuna l'accesso alle strutture dati avviene tramite JavaScript
 - Di facile inibizione
 - Ma se mi serve?



Custom Protocol Handlers

- HTML5 prevede la possibilità per un sito web di registrarsi come content handler per protocolli o MIME types
- Il browser tratterà tale sito come applicazione per aprire il protocollo o tipo di file selezionato
- Semplice, no?



Custom Protocol Handlers

- L'applicazione si registra per un MIME type molto comune (esempio: image/jpg)
- L'applicazione lascia un cookie univoco per ogni visitatore
- La vittima visita l'applicazione durante una sessione Tor
- La vittima visita l'applicazione durante una sessione non-Tor
- ...
- Profit!



Offline Web Applications

- HTML5 prevede la possibilità di salvare su disco le applicazioni web e potervi interagire anche in assenza di connettività a Internet
- Regole stringenti per l'installazione e la rimozione di queste applicazioni
- La tecnologia si sta diffondendo grazie a Google Gears (che fa anche molto altro in più)



Offline Web Applications

- Tecnologia strettamente basata su javascript: si applicano tutti i vettori di attacco noti
- Leak di privacy se non si effettua un oculato passaggio di stato tra online/offline
 - Il client scarica l'applicazione in una sessione Tor
 - Il client usa l'applicazione offline
 - Il client torna online con una sessione non-Tor



Multimedia: <embed>, <object>

- Permettono di includere contenuti multimediali all'interno della pagina
 - L'attributo “src” determina l'url della risorsa inclusa
 - L'attributo “type” permette di richiedere l'utilizzo di determinati plugin o handler associati nel browser
- <embed> ha qualche restrinzione in più rispetto a <object>



Multimedia: <video>, <audio>, <source>

- Definiscono una risorsa multimediale contenuta in una pagina
- Permettono il controllo del playback della risorsa via JavaScript
- <source> si comporta come <embed> e <object>, usando gli attributi src e type



Multimedia

- Se vengono abilitati i JavaScript per poter fruire dei contenuti, si cade nelle tipologie già viste di attacco
- `<embed>`, `<object>` e `<source>` permettono di lanciare applicazioni registrate presso il MIME type dichiarato, un ottimo metodo per aprire side channel



Difese

- TorButton protegge in parte la navigazione con Firefox
 - Isolamento sessioni
 - Isolamento cookie jar
- Impedire l'utilizzo di content-handler registrati sotto Tor durante sessioni non-Tor e viceversa
 - Bloccare i content-handler durante le sessioni Tor?
- Impedire il ritorno online di una web-application in una sessione di tipo differente da quella di partenza



Geo Location

- Possibilità di avere applicazioni definite location-aware
- I dati vengono presi da:
 - Informazioni di cella wifi
 - Servizio originale di Loki.com
 - Firefox 3.5 usa uno scambio di cookie bisettimanale con i servizi di Google
 - Dispositivi GPS eventualmente collegati
 - GeoIP sembra fuori moda o poco pratico



Geo Location

- In Firefox l'invio delle informazioni è a scelta dell'utente
- Le api per accedere alla posizione del client sono in JavaScript
- Per fortuna HTML si occupa solo di presentazione e HTML5 non specifica ancora nessuna richiesta in merito ;-)



Eye Candy

- Avete mai provato GetPersonas?
- Un sistema di skin “leggere” per Firefox
 - Personalizzano il background delle barre degli strumenti in testa al browser
 - ... e della status bar in fondo
 - Limite massimo di 300KB a skin
- Progetto ufficiale di Mozilla.org
- Piuttosto carine, lo ammetto!



Eye Candy

- L'estensione contatta il server delle skin di Mozilla per permettere cambiamenti immediati dei background
- Il sito lascia al client un cookie contenente l'indirizzo IP :(
- Il cookie non viene cambiato immediatamente al cambio di indirizzo :(
 - Per un exit node che controlla il traffico verso il server Personas è facile leggere l'IP registrato dal cookie



Conclusioni

- JavaScript
 - La “colla” del Web2.0
 - Offre grandissime potenzialità per creare applicazioni web ricche e interattive...
 - ...e per mettere a rischio la privacy degli utenti Tor (e non solo)
- HTML5
 - Introduce interessanti aree di attacco
 - Permette più facilmente la creazione di side-channel per bypassare Tor



Webografia

- <http://trivero.secdiscover.com/html5whitepaper.pdf>
- <http://trivero.secdiscover.com/moca.pdf>
- <http://dev.w3.org/html5/spec/Overview.html>
 - Articolo 5.8.2 (Custom Content Handler)
 - Articoli 5.9 e seguenti (Offline Web Applications)
 - Articoli 4.8.4-9 (Multimedia)
- https://developer.mozilla.org/En/Using_geolocation
- <http://www.getpersonas.com/>



Domande?

