SafeBreach

Stop tomorrow's breach.
Today.

In Plain Sight:
The Perfect Exfiltration

Itzik Kotler
Amit Klein

Help -> About -> **Itzik Kotler**

- 15+ years in InfoSec
- CTO & Co-Founder of Safebreach
- Presented in RSA, HITB, BlackHat, DEFCON, CCC, …
- http://www.ikotler.org

Help -> About -> **Amit Klein**

- 25 years in InfoSec
  - **VP Security Research, Safebreach** 2015-present
  - CTO Trusteer (acquired by IBM) 2006-2015
  - Chief Scientist Cyota (acquired by RSA) 2004-2006
  - Director of Security and Research, Sanctum (now part of IBM) 1997-2004
  - IDF/MOD (Talpiot) 1988-1997
- 30+ papers, dozens of advisories against high profile products
- Presented in RSA, CertConf, BlueHat, OWASP, AusCERT, ….

Goal

Endpoint (desktop) in a high security enterprise needs to exfiltrate data to an entity outside the enterprise.

The required bandwidth can be very modest - up to several dozen bits per day.

Example use cases:

- Cryptographic keys (128 bit)
- Yes/no data e.g. “war alert”
- Closed set data (“who is the mole”), names, short sentences (few words) e.g. strategic decisions

The Rules of the Game

The 10 commandments, part I

1. **Scalability and Security:** A perfect exfiltration technique should adhere to Kerckhoffs's principle (sometimes called Shannon's maxim), which states that "A [crypto] system should be secure even if everything about the system, except the key, is public knowledge". In other words - no "security by obscurity". **This enables multiple channels with different keys**

2. **Only web (browsing) and derived traffic is allowed (e.g. HTTP, DNS, TLS)**

3. "Whenever there is any doubt, there is no doubt" - anything that may theoretically be perceived as passing info is forbidden. So no emails, no encrypted texts, no google docs, no forum posts, etc.

The 10 commandments, part II

4. Perfect network monitoring. Every packet is scrutinized at all protocol levels, every anomaly is detected (“big data/machine learning, etc.”), reputation and statistics for all IPs/hosts are available, etc.

5. Assume TLS/SSL termination at the enterprise level

6. Receiving party has no restrictions (e.g. network surveillance)

7. No nation-state monitoring or 3rd party site monitoring. Though it’s assumed they do have some basic IDS/IPS/security products, so avoid flooding, etc.

The 10 commandments, part III

8. Time synchronization (seconds-resolution) between the communicating parties

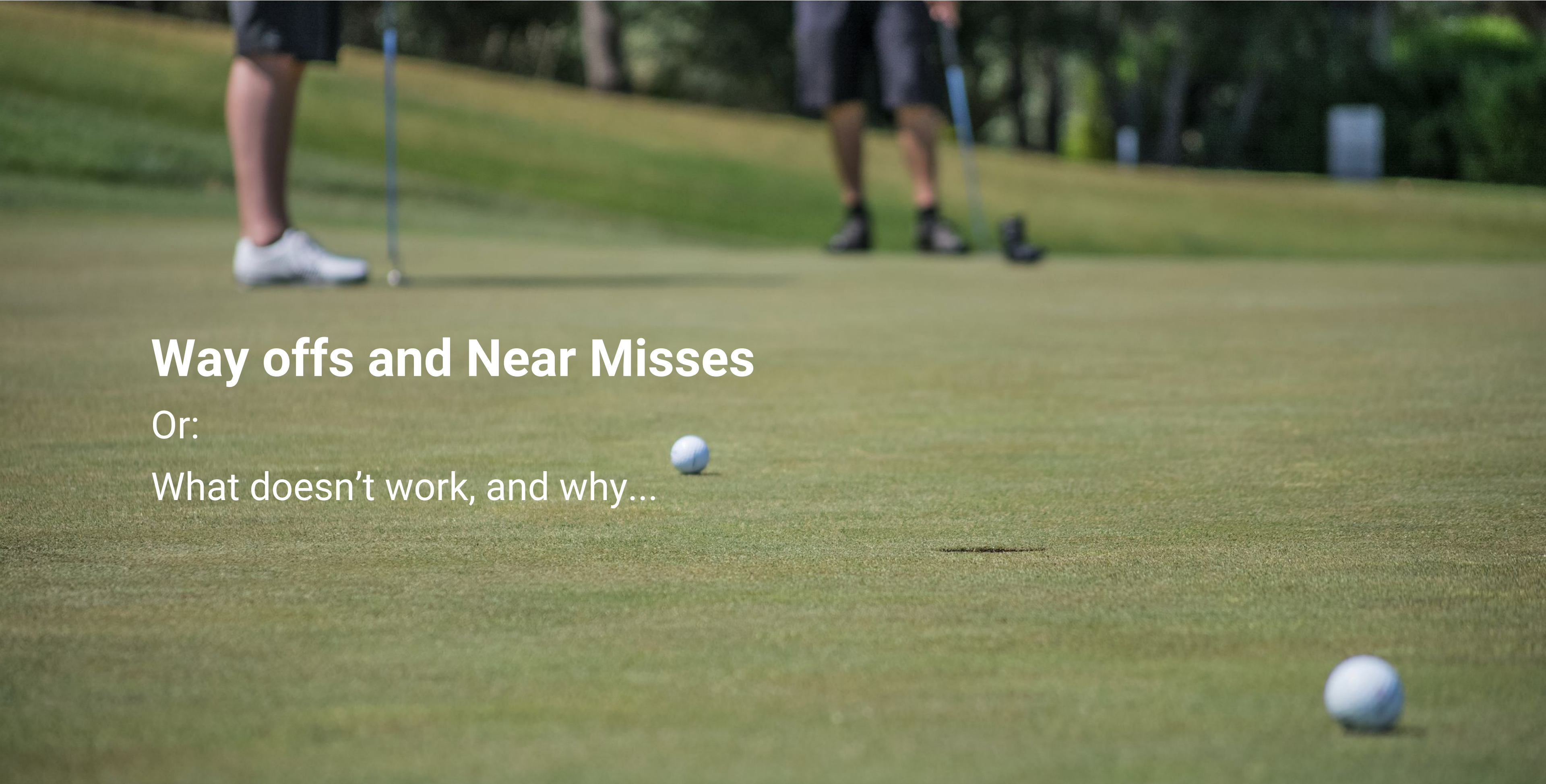
9. Bonus points for methods that can be implemented manually (without SW) at the sender side

10. Active disruption by the enterprise is an option

Commandment #4 -> **No direct communication**

Commandment #4: Perfect network monitoring. Every packet is scrutinized at all protocol levels, every anomaly is detected (“big data/machine learning, etc.”), reputation and statistics for all IPs/hosts are available, etc.

Since site/IP ownership can be determined, and since we assume reputation and statistics are available to the monitoring entity, it follows that we cannot use direct communication.
The covert channel has to be indirect (i.e. through a 3rd party site not owned by the communicating entities).



Way offs and Near Misses

Or:

What doesn’t work, and why...

Way off - **“Traditional” covert channels**

Many techniques (fields in IP/TCP/HTTP) requiring direct communication

- TCP/IP: TTL, packet length, timestamp, packet rate/timing
- HTTP: If-Modified-Since, Cookie, whitespaces, etc.

Comprehensive lists:

http://eprints.ugd.edu.mk/10284/1/surveyAMBPselfArc.pdf

http://caia.swin.edu.au/cv/szander/publications/szander-ieee-comst07.pdf

http://docslide.us/internet/covert-timing-channels-using-http-cache-headers.html

Way off example

TOS header (IP header)

- 8-bit IP header field, can be used to exflitrate data
- De-facto Windows workstations send TOS=0.
  - Anomaly detection is thus trivial, especially with prior knowledge of the concept
  - Disruption is trivial - set TOS=0 at firewall for all outgoing packets
- Needs direct connection to the 2nd party
  - No good.

Near miss - IP ID (indirect)

IP ID covert channel (http://www0.cs.ucl.ac.uk/staff/G.Danezis/papers/cover.pdf)

- 3rd party using OS with global incrementing IP ID. Same concept as idle scan. Sender sends a request to the 3rd party site, the answer from the site increments the IP ID counter. The receiver observes (before vs. after). One bit per packet (sent/not sent).
  INDIRECT
- The only OS that provides globally incrementing IP ID these days is FreeBSD. And in FreeBSD 11.0 (to be released July 2016) the TCP layer sets DF=1, IPID=0. So can't be used with web browsing (TCP).
- Line noise

Very near miss - web counters

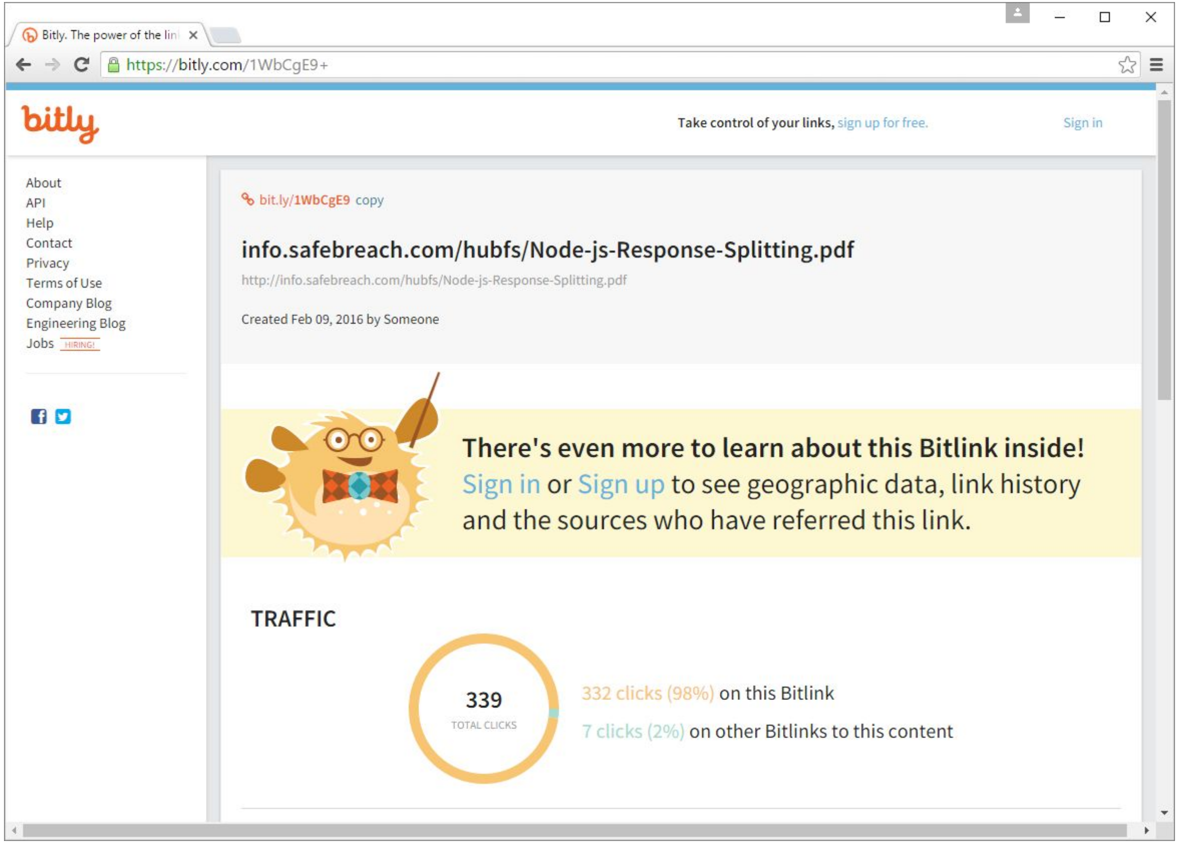
Bit.ly (and probably other URL shorteners) - based.
Sender and receiver agree on a bit.ly URL, and time.

Needs to be a not-so-popular resource (but not too rare), Ideally a URL that is linked from the Internet (e.g. a tweet/blog).

Protocol:

- Before designated time, receiver checks web count for the URL (in bit.ly, appending a “+” to the URL).
- On time, sender requests the URL (1) or not (0).
- A bit later, the receiver checks the web count again.

Bit.ly web counter example



Bitly. The power of the lin
https://bitly.com/1WbCgE9+
bitly
Take control of your links, sign up for free.
Sign in
About
API
Help
Contact
Privacy
Terms of Use
Company Blog
Engineering Blog
Jobs HIRING!
copy

bit.ly/1WbCgE9

info.safebreach.com/hubfs/Node-js-Response-Splitting.pdf

http://info.safebreach.com/hubfs/Node-js-Response-Splitting.pdf

Created Feb 09, 2016 by Someone

There's even more to learn about this Bitlink inside! Sign in or Sign up to see geographic data, link history and the sources who have referred this link.

TRAFFIC

339 TOTAL CLICKS

332 clicks (98%) on this Bitlink

7 clicks (2%) on other Bitlinks to this content

Attack (disruption)

Intercept every request for a bit.ly URL,
send a request to the URL+,
obtain the original URL and redirect to it internally
(without going through bit.ly and incrementing the counter).

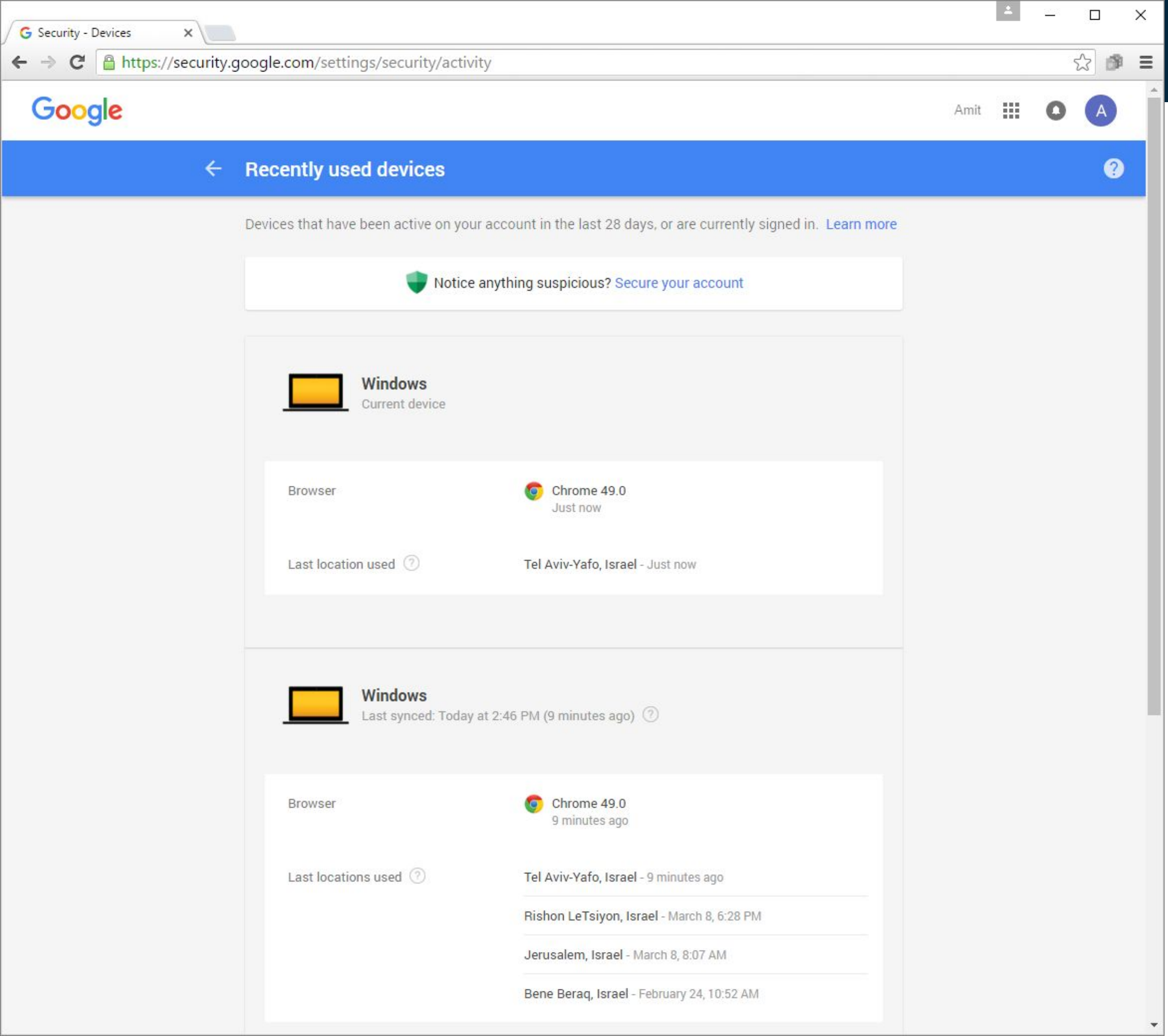
Near miss - account login time

Gmail shows last login time (https://security.google.com/settings/security/activity).
So the protocol is that the sender is logged in at a designated time (bit=1) or not (bit=0).
Receiver can login later and use the URL to see when the sender logged in (= covert channel)

BUT - the page also shows last access from all devices, with browsers and geo-location.
So the enterprise can request this page as well and observe.
Yahoo mail - also shows IP address

Problem

- In general, some enterprises block email/social sites.



Recently used devices

Devices that have been active on your account in the last 28 days, or are currently signed in. Learn more

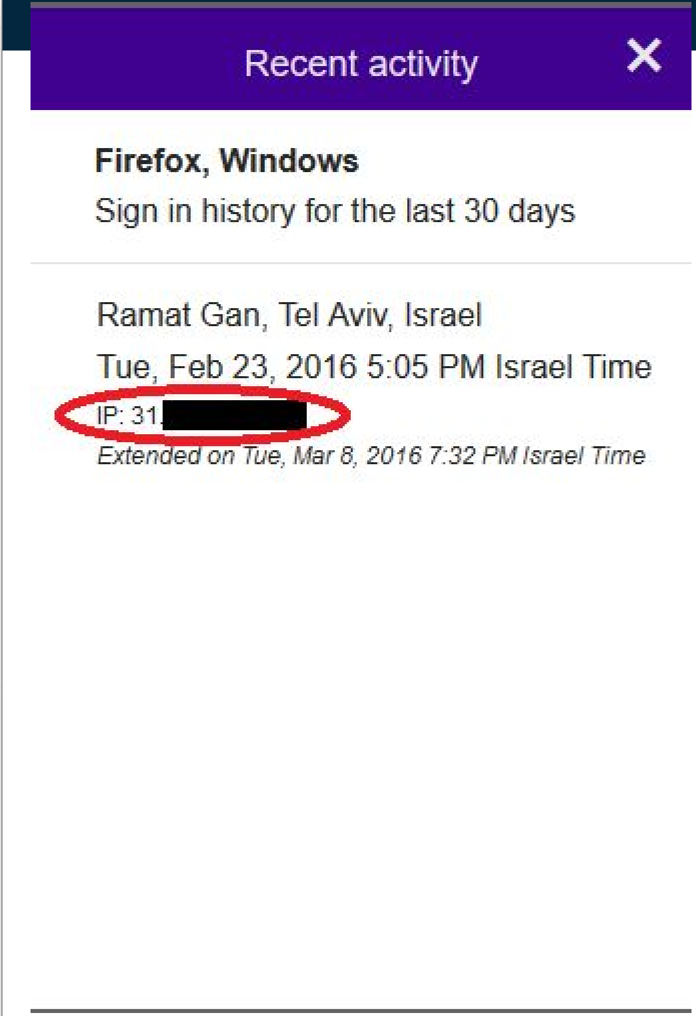
Notice anything suspicious? Secure your account

Windows
Current device

Browser	Chrome 49.0 Just now
Last location used	Tel Aviv-Yafo, Israel - Just now

Windows
Last synced: Today at 2:46 PM (9 minutes ago)

Browser	Chrome 49.0 9 minutes ago
Last locations used	Tel Aviv-Yafo, Israel - 9 minutes ago Rishon LeTsiyon, Israel - March 8, 6:28 PM Jerusalem, Israel - March 8, 8:07 AM Bene Beraq, Israel - February 24, 10:52 AM



Recent activity

Firefox, Windows

Sign in history for the last 30 days

Ramat Gan, Tel Aviv, Israel

Tue, Feb 23, 2016 5:05 PM Israel Time

IP: 31[redacted]

Extended on Tue, Mar 8, 2016 7:32 PM Israel Time

The Perfect Exfiltration

The principles of perfect exfiltration

Use regular browsing (HTTP/HTTPS)

Modify the application (or higher protocol) state

- Storage channel

Observe the change remotely

HTTP server-side caching

- Find a popular site that caches pages on-the-fly.
  - Has to have tons of pages
  - Has to have pages that are not so popular
  - => eCommerce sites are ideal.
- Typically the page’s caching time can be deduced from the HTTP response headers (Expires, etc.)
- Sender and receiver agree on a page (URL) and a time.
  The page has to be non-popular.
- To send 0, sender does nothing. To send 1, sender makes an HTTP request for the page at the prescribed time.
- 10 seconds after the agreed upon time, the Receiver makes an HTTP request for the URL, and observes whether the page was cached recently (1),
  or right now (0).

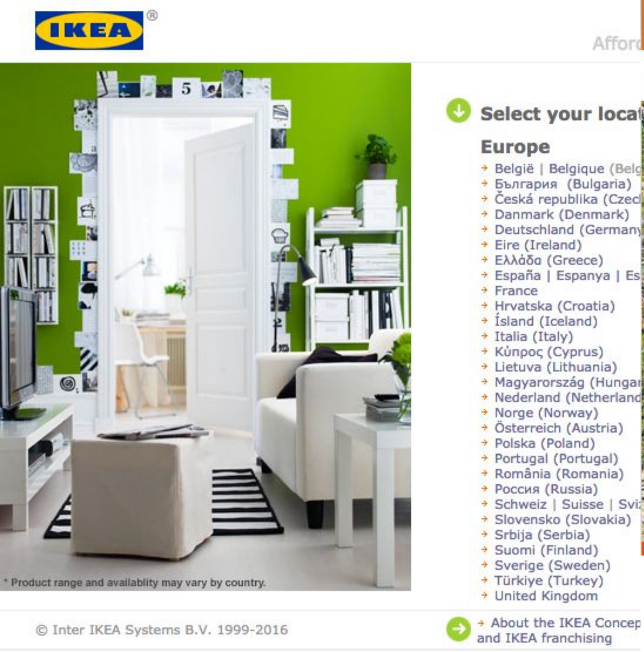
SafeBreach

Example sites

ikea.com

easyjet.com

zap.co.il



Select your locat

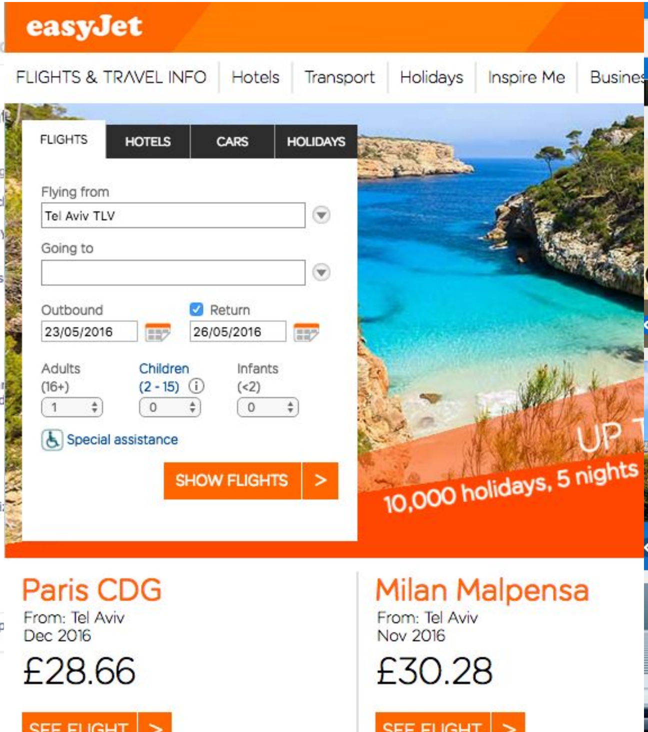
Europe

- België | Belgique (Belg
- България (Bulgaria)
- Česká republika (Czec
- Danmark (Denmark)
- Deutschland (German
- Eire (Ireland)
- Ελλάδα (Greece)
- España | Espanya | Es
- France
- Hrvatska (Croatia)
- Ísland (Iceland)
- Italia (Italy)
- Κύπρος (Cyprus)
- Lietuva (Lithuania)
- Magyarország (Hunga
- Nederland (Netherland
- Norge (Norway)
- Österreich (Austria)
- Polska (Poland)
- Portugal (Portugal)
- România (Romania)
- Россия (Russia)
- Schweiz | Suisse | Svi
- Slovensko (Slovakia)
- Srbija (Serbia)
- Suomi (Finland)
- Sverige (Sweden)
- Türkiye (Turkey)
- United Kingdom

* Product range and availablity may vary by country.

© Inter IKEA Systems B.V. 1999-2016

About the IKEA Concep and IKEA franchising



Flying from: Tel Aviv TLV

Going to:

Outbound: 23/05/2016

☑ Return: 26/05/2016

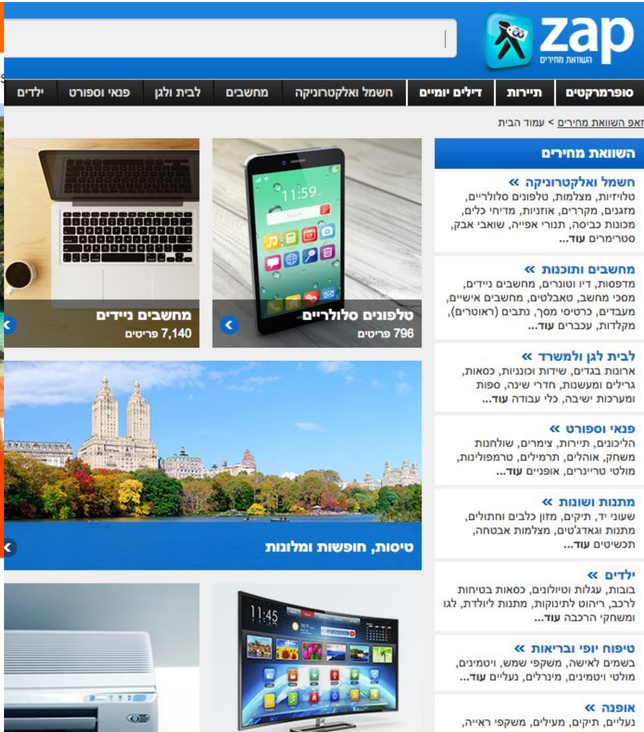
Adults (16+): 1

Children (2 - 15): 0

Infants (<2): 0

Paris CDG
From: Tel Aviv
Dec 2016
£28.66

Milan Malpensa
From: Tel Aviv
Nov 2016
£30.28



זאפ השוואת מחירים > עמוד הבית

השוואת מחירים

חשמל ואלקטרוניקה »
טלויזיות, מצלמות, טלפונים סלולריים, מזגנים, מקררים, אוזניות, מדיחי כלים, מכונות כביסה, תנורי אפייה, שואבי אבק, סטרימרים **עוד...**

מחשבים ותוכנות »
מדפסות, דיו וטונרים, מחשבים ניידים, מסכי מחשב, טאבלטים, מחשבים אישיים, מעבדים, כרטיסי מסך, נתבים (ראוטרים), מקלדות, עכברים **עוד...**

לבית לגן ולמשרד »
ארונות בגדים, שידות וכונניות, כסאות, גרילים ומעשנות, חדרי שינה, ספות ומערכות ישיבה, כלי עבודה **עוד...**

פנאי וספורט »
הליכונים, תיירות, צימרים, שולחנות משחק, אוהלים, תרמילים, טרמפולינות, מולטי טריינרים, אופניים **עוד...**

מתנות ושונות »
שעוני יד, תיקים, מזון כלבים וחתולים, מתנות וגאדג'טים, מצלמות אבטחה, תכשיטים **עוד...**

ילדים »
בובות, עגלות וטיולונים, כסאות בטיחות לרכב, ריהוט לתינוקות, מתנות ליולדת, לגו ומשחקי הרכבה **עוד...**

טיפוח יופי ובריאות »
בשמים לאישה, משקפי שמש, ויטמינים, מולטי ויטמינים, מינרלים, נעליים **עוד...**

אופנה »
נעליים, תיקים, מעילים, משקפי ראייה,

Special Care and Caveats (part I)

- Sites do change. Ideally have a downlink to the sender to update URLs

- Time sync - ideally seconds-resolution. The better the sync, the less the receiver needs to wait before reading (up to round trips). Noise can be introduced if waiting for too long. Further research needed to measure and advise on thresholds.

- On the other hand, the longer the time between the sender checking in and the receiver polling, the more secure the scheme becomes against attacks (see next slide)

Special Care and Caveats (part II)

- Resilience - send “01” or “10” (to detect network outages, etc.)
- Need to reconstruct the complete browsing session in order to blend in
- Immerse in many regular web traffic sessions to dilute the real sessions
- Multiple cache servers (not encountered), geo-ip based routing/distribution

(Failed) Attack example

Intercept every user’s requests.

Hold and delay every user’s request by e.g. 11 seconds

Observe whether the page was cached (due to the receiver’s hit) in the last 11 seconds.

Both detection and disruption

Cons:

- Very noisy - horrible user experience…
- What does “positive” mean?

How do we fare vs. the 10 commandments?

#1 - Secure and scalable. We can assign a unique URL (possibly in a different site) to a bit. Knowing the method, and even previous URLs doesn't help (our best effort attacks are not convincing)

#2 - web only. Yup ;-)

#3 - no doubts. The traffic does look 100% kosher, no sender bits are encrypted/deposited, all traffic can be explained and reconstructed to show no hidden data.

#4 - perfect network monitoring - all traffic originates from the browser and destines at a popular site, can be attributed to regular activity

How do we fare vs. the 10 commandments?

#5 - TLS/SSL termination. Not a problem.

#6 - receiving party has no restrictions. We’re OK with that…

#7 - no nation state surveillance - we’re OK with that too + no impact on the site.

#8 - Time sync (sec. resolution) - we’re OK with that.

#9 - can be generated manually (web browsing)

#10 - active disruption - impractical

Demo Time!

```
git clone https://github.com/SafeBreach-Labs/cachetalk.git
cd cachetalk
python cachetalk.py -h
```

Web counters (take II)

- Counting specific page/resource
  - YouTube movies, StackOverflow topics, etc.
- Sender and receiver agree on a resource (not too popular) and a time.
- **Receiver can observe the counter (before and after the designated time) without modifying the counter**
- Sender downloads the resource (1) or not (0) at a designated time.

Problems:

- Some social sites (e.g. youtube) can be blocked by some enterprises
- The “story” may be challenging

“online “ status (chats)

Web-based chat services (e.g. facebook, google talk, yahoo chat, skype) typically have “online status” for friends.

Each party opens an account. Then they befriend each other.
Sender is online (1) or offline (0) at the designated time.

Attacks/problems

- Social sites may be blocked
- The initial befriending can be questioned

Summary and Conclusions

Perfect exfiltration exists!

Robust “commandments” (derived from attack model) can be used to evaluate candidate protocols

So far, only low bandwidth protocols were discovered

Challenge (1): find higher bandwidth perfect exfiltration protocols

Challenge (2): find ways to defeat the (perfect) exfiltration protocols described

Q&A

Amit Klein <amit.klein@safebreach.com>

Itzik Kotler <itzik.kotler@safebreach.com>