

KRESTEN JACOBSEN

---

# HTTPS (OG SSLSTRIP)

## BAGGRUND: HTTPS – HYPERTEXT TRANSFER PROTOCOL SECURE

- ▶ **Leverer HTTP sikkert\* ud til klienter**
  - ▶ Autentificering
  - ▶ Integritet
  - ▶ Konfidentialitet
- ▶ **Teknisk implementering**
  - ▶ TLS (tidligere SSL).
  - ▶ Data krypteres i transit (server <==> klient).

Autentificering: Er den der tager telefonen, den man har ringet op?

Integritet: Bliver det vi siger til hinanden transmitteret korrekt og uændret?

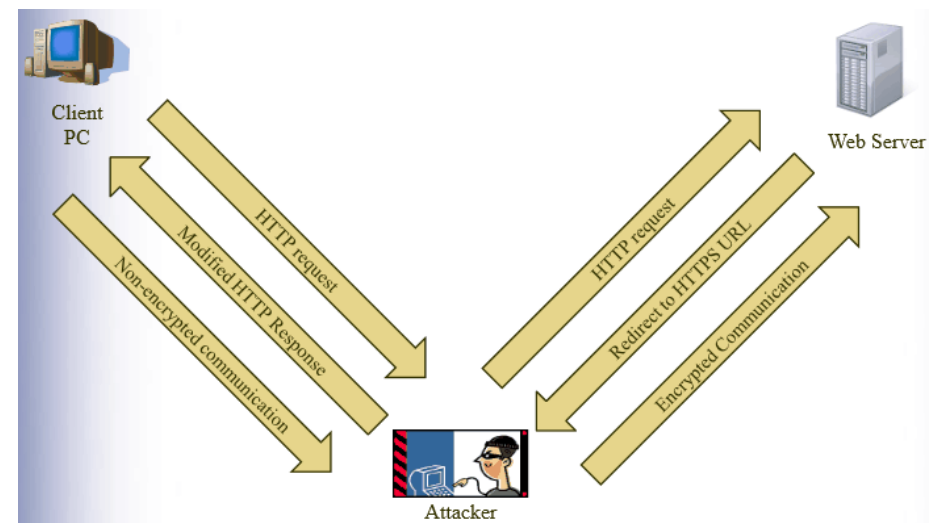
Konfidentialitet: Bliver det vi siger overhørt af andre?

TLS (tidligere SSL).

Data krypteres i transit (server <==> klient).

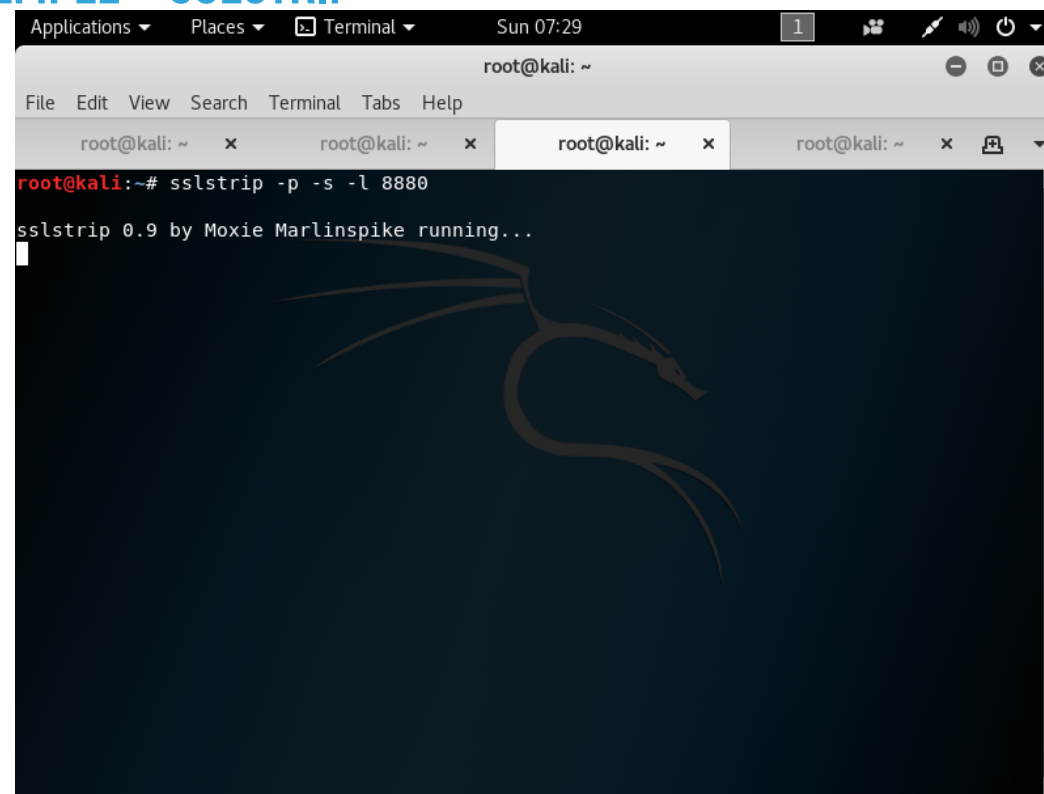
## BAGGRUND: HTTPS – HYPERTEXT TRANSFER PROTOCOL SECURE?

- ▶ \* Bortset fra Man In the Middle-angreb



Som her faktisk er et 'protocol downgrade attack', fordi vi 'downgrader' HTTPS-forbindelsen til HTTP.

## EKSEMPEL – SSLSTRIP

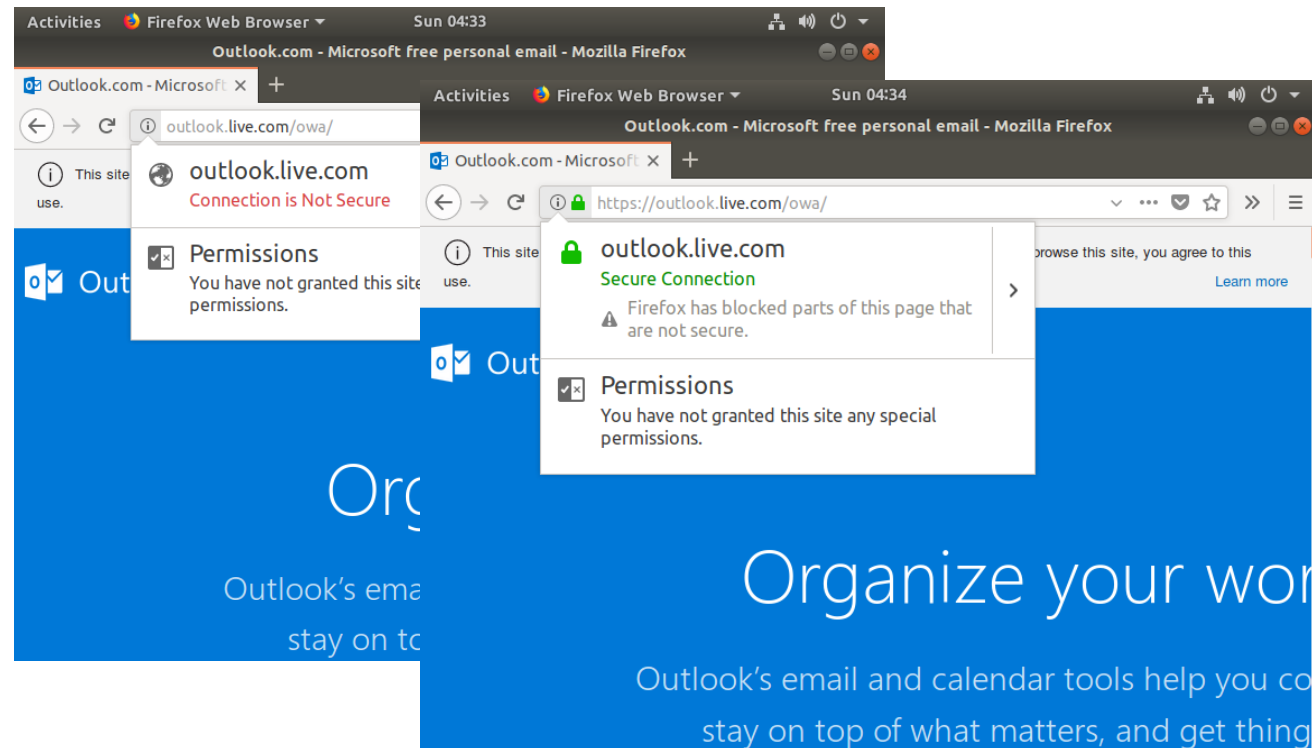


```
root@kali: ~  
File Edit View Search Terminal Tabs Help  
root@kali: ~ x root@kali: ~ x root@kali: ~ x root@kali: ~ x  
root@kali:~# sslstrip -p -s -l 8880  
sslstrip 0.9 by Moxie Marlinspike running...  
[Kali Linux Dragon Logo]
```

**Udførsel:**

- 1) Lav MiTM-angreb (brug ARP-poisoning; se slides herom).
- 2) Start SSLStrip: `sslstrip -p -s -l 8880`

## EKSEMPEL - SSLSTRIP

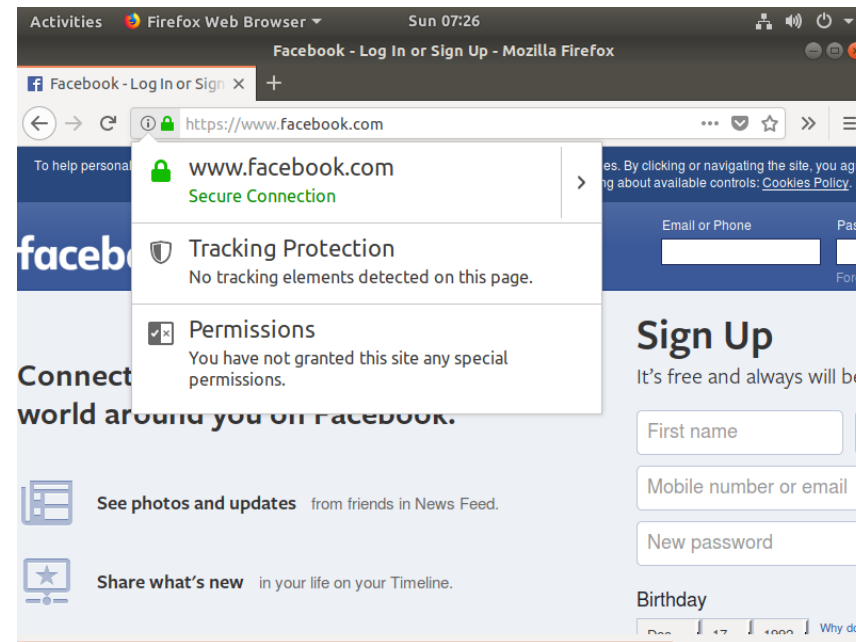


På den angrebne:

- 1) Besøg [hotmail.com](https://hotmail.com)
- 2) Tjek sslstrips log, hvor al trafik kan ses i klar tekst.

# SSLSTRIP – BEGRÆNSNINGER

## ► HSTS...



HSTS er en 'policy', der angiver at en klient - i et fast defineret tidsrum (typisk ikke under et år) - kun må kommunikerer sikkert med serveren, hvorfor et downgrade attack bliver noget sværere at udføre.

Populære browsere kommer med en *foruddefineret* liste over væsentlige websites, som skal afvikles med HSTS.