

# Client Zertifikate

– Aus Alt mach Neu –

CAcert - Die Community CA

Basiert auf einen Vortrag von Ian Grigg



## Login v0.0 bis ...

- Login 0.0: Jedem wird vertraut
- Login 0.1: Passworte + Usernamen
- Login 0.3: Single Sign On (SSO) – Der Traum
- Login 0.4: Verbund ...

# Was lief schief ?

- 0.0 Vertrauen
- 0.1  $N * \text{Komplexität} \neq \text{Support} + \text{Sicherheit}$
- SSO

...

# Was lief schief ?

- SSO
  - a) Jede Website, hat eine andere Zugriffsmethode (Bedienung, Buttons) ... (Henne)
  - b) Jede Person, eine Methode (von unterschiedlichen Diensteanbietern) ... (Ei)
  - c) Wer kommt an meine Daten ?
  - d) Wer ist hier Kunde ?

# Einsatz von Computern ?

- Haben wir dafür eigentlich nicht Computer um mit solchem Zeugs umzugehen ?

# Einsatz in Computern

- Wir haben !
- Das “Zeugs” nennt sich “Client Zertifikate”
  - Öffentliche – Private Schlüssel Paare
  - 3<sup>rd</sup> Party Signaturen

*Third Party Signaturen sind Assurances über einen Notar. Assurance ist möglich in Ländern mit geringer Assurer Dichte. Derzeit aber ist das Verfahren aufgrund fehlender Subpolicy auf Eis gelegt*

# Einsatz in Computern

- Wir haben !  
...
- Sicher, sie sind wie “Crypto” Passworte
- Funktionieren mit jedem Browser, Webserver
- Warum haben sich Client Zertifikate nicht etabliert ?

# Warum haben sich Client Zertifikate nicht etabliert ?

- Eine unendliche Anzahl unterschiedlichster Software
- Daten sind keinem Risiko ausgesetzt auch nicht die Benutzer
- Stellt ein Gegengewicht zu der gängigen Zugangsmethode Account/Passwort dar ... lässt sich nicht skalieren
- Henne & Ei: Niemand hat ein Ei
- (Frage nicht)

# CAcert steigt in Das Eier Geschäft ein

- Zertifikate => “Identität” => Assurance
  - Web Of Trust
- Audit!
  - Wie auditiert man einen Web Of Trust ?
  - Dokumentation ... Standards ...  
Überprüfbarkeit ...
- CATS – CAcert Automated Testing System
  - Alle Assurer müssen sich einem Test unterziehen

# Inspiration!

- CATS benötigt Client Zertifikat (kein Passwort)
  - Weil wir eine CA sind ?
  - So das unsere Assurer über Zertifikate bescheid wissen ?
  - Wir wollen Cool aussehen ?
  - Wir wünschen High-Security Zugang ?
  - Oder ?
  - Frage nicht ...

# Der Erfolg von CATS

- Start Anfang 2008
- Obligatorisch seit Anfang 2009
- 10K++ → 1000 → 2000 → 3000
  - Heute 3580
  - Faustregel: seriöser Test Reduzierung auf 1/3
- Assurer Gemeinschaft ist stärker

# CAcert steigt in das Hühnergeschäft ein

- Jeder Assurer hat ein Zertifikat
- Daher ... Jede Website kann Zertifikats Zugang bereitstellen (nur Zertifikats Zugang)
- Migriere alle Websites zu ausschliesslichem Zertifikatszugang
- Wordpress, Sympa, Voting ... abgeschlossen
- Steht auf den Sysadmins ToDo Listen

# Ergebnisse ... für den Blog

- Schreibzugriff, wenn du ein Client Zertifikat hast
  - Mehr Autoren, mehr Artikel ...
- Spam Problem gelöst
- Nie mehr vergessene Accounts  
falsche Passworte
- → Administrator kann sich um andere Dinge kümmern

# Ergebnisse ... für den Blog (Fortsetzung)

- Keine längeren Diskussionen über das WER schreibt den Artikel
- User können sich mehr Zeit nehmen und auf den Artikel verwenden

# Problemfälle!

- #1 Mehrere Zertifikate → Firefox Konfusion
  - Wir warten auf das User-Whitelisting
- #2 Verrückte Meldungen
  - Server weist Zertifikat ab
  - Client erzählt, Server lehnte Handshake ab
  - Benutzer lehnt alles ab
  - Entwickler sind sich nicht einig
  - (warten auf noch mehr User Beschwerden)

# Strategien

- Hybrid: Passwort + Zertifikatszugang
  - Wenn du musst ...
  - (CA Hauptseite hat dies, für die Wiederherstellung)
- Nur Zertifikate – Immer Zertifikate:
  - a) Apache Abwicklung (zu wenig, zu viel)
  - b) Applikations Abwicklung (du musst programmieren)
- Empfehlung: Nur Zertifikate – via Applikationen

# Beispiel: Apache

- Basic Client Side Authentizierung für den Fall Zugang nur mit Zertifikaten. Jedes beliebige Zertifikat aus einem Set von CA's

```
## Client Verification
SSLVerifyClient optional
SSLVerifyDepth 3
SSLCADNRequestPath /usr/share/ca-certificates/cacert.org/

# error handling
RewriteEngine on
RewriteCond %{SSL:SSL_CLIENT_VERIFY} !=SUCCESS
RewriteRule .? - [F]
ErrorDocument 403 "You need a client side certificate issued by CACert to access this site"
```

# Beispiel: Apache (Fortsetzung)

- SSLCADNRequestPath enthält den Pfad der Zertifikate, die akzeptiert werden sollen

```
drwxr-xr-x  2 root root  4096 2009-05-14 23:22 .
drwxr-xr-x  9 root root  4096 2007-05-16 23:12 ..
lrwxrwxrwx  1 root root     8 2009-05-14 23:22 5ed36f99.0 -> root.crt
-rw-r--r--  1 root root  2151 2007-03-04 05:23 class3.crt
lrwxrwxrwx  1 root root    10 2009-05-14 23:22 e5662767.0 -> class3.crt
-rw-r--r--  1 root root  2569 2007-03-04 05:23 root.crt
```

- Details unter  
<http://wiki.cacert.org/ClientCert>

# Beispiel: PHP

- Abfrage mit Hilfe von Systemvariablen

```
if ($_SERVER[SSL_CLIENT_VERIFY] == "SUCCESS")
{
    $sql = "SELECT * FROM " . USERS_TABLE . "
        WHERE user_email = " .
        $db->sql_escape($_SERVER[SSL_CLIENT_S_DN_Email]) . """;
    $result = $db->sql_query($sql);
    $row = $db->sql_fetchrow($result);
    $db->sql_freeresult($result);
    if ($row)
    {
        return $row;
    }
}
```

# Eingehende Strategie



- Problemfall #3 – Zertifikate können und werden sich ändern
- Lese Zertifikat in die Datenbank ein
  - Zertifikats Index → Account Zuordnung
- Für neue Zertifikate, Scan einiger Infos
  - kann auf Email und Name prüfen
- Wenn User Name und Email ändert ...
  - bedarf es weiterer Überlegungen

# Zusammenfassung



- Zertifikate verrichten Dienste
  - wesentlich besser als Passworte
  - weniger Stress wenn es mal funktioniert
  - weniger Arbeit für Administratoren
- Gegenüber anderen Methoden ?
  - Höhere Sicherheit als OpenID
  - Verfügbar (sobald du ein paar Eier gekauft hast)

# Herausforderung



- Problem (b): Niemand bekommt ein Ei
- Herausforderung für dich: Verteile Zertifikate an alle User
  - “alle sind CACert”
  - Erstelle eine Site, jede Seite, benutze Client Zertifikate
  - Intern: benutze die Zertifikate vom Hersteller

# Dank, Fragen, Antworten

- Noch Fragen ?
- <http://www.cacert.org>
- <http://wiki.cacert.org>
- Ulrich Schroeter  
[ulrich@cacert.org](mailto:ulrich@cacert.org)

