Trusted Computing

Daniel Bosk[1]

Department of Information and Communication Systems (ICS),
Mid Sweden University, Sundsvall.

trustcomp.tex 2068 2014-11-03 10:52:07Z danbos

Overview

Mittuniversitetet
MID SWEDEN UNIVERSITY

Overview

Mittuniversitetet
MID SWEDEN UNIVERSITY

What is DRM?

- The main purpose of DRM is to prevent piracy.
- This can be applied to all sorts of material; from photos, to films, to application programs, and all the way to operating systems.
- There are different approaches and purposes, e.g. to control piracy, but also to control the selling of used products.

Historical Approaches

- In the dawn of computing software was given away for free by the hardware (HW) vendors.
- This was one way to promote sales of HW, the users needed software to use the HW.
- This changed, and in the 1960's software was a significant cost.
- Now HW vendors charged extra for their OSes and there were third-party software vendors.

Historical Approaches

- In the 1970's software could be turned into general packages.
- I.e. software needed no longer be customised to the users' HW.
- Now problems with the ownership of code rose, what if one of your programmers left for a competitor and their program soon got some of your features.
- To determine if the programmer copied the source or reinvented it, software birthmarks could be used – i.e. analysing how the software is coded.

Historical Approaches

- Then came the 1980's, with these general purpose computer systems came attempts at copyright enforcement.
- Some approaches was to lock the software with an error message every few months, e.g. "Error X: Please call technical support", where X is a customer specific number.
- This worked for as long as users were technically unknowledgable and it didn't cross the limit what was considered reliable.
- Other apporaches was for the software to look at the processor's serial number.

Historical Approaches

- In summary, there was essentially three general approaches tried.
- First, to add uniqueness to the machine; e.g. a HW dongle.
- Second, to create uniqueness within it; e.g. install the software in a way that prevented naïve copying (cf. Adobe Photoshop which modified the boot loader and accidentally removes Grub).
- Generally people must be able to create a backup, but not copy those backups for sharing (copy generation control).
- And third, to use whatever uniqueness there already was; e.g. storing the characteristics of the computer, cards present, amount of memory, etc.
- This approach needs to handle HW upgrades though.

Modern Approaches

- One of the more modern approaches is to have the software connect to the vendor's servers to verify itself.
- This works as long as the software isn't needed offline.
- But even online it can be really annoying, cf. Ubisoft's Assassin's Creed DRM which required a constant connection.
- Another is to leave some critical part to be done by the vendor's servers.
- An example of this is Blizzard's Diablo 3 games, which lets the server handle the entire game (map generation, NPCs, etc.).

Modern Approaches

- Yet other approaches is to encrypt vital parts, e.g. some code or video.
- This can be used for both software and media, for which it is popular (DVD, BlueRay, streaming services).
- However, this must be decrypted before use ...

Overview

Mittuniversitetet
MID SWEDEN UNIVERSITY

What is a protocol?

- Ett system består av en uppsättning principals.
- Ett protokoll är en uppsättning regler som styr hur dessa kommunicerar.

Example (Tentamen MIUN)

1. Tentamensvakten öppnar salen och ger varje tentand ett nummer.
2. Tentanden går in och sätter sig vid sin tilldelade plats.
3. Efter att tentan börjat jämför tentamensvakten tentandens legitimation och nummer.
4. Vid inlämning av skrivning jämförs legitimationen och numret.

What is a protocol?

- Bör vara designade för att motstå attacker.
- Både oavsiktligt och avsiktligt brott mot protokollet.

What is a protocol?

- Konstrueras utifrån grundläggande antaganden.
  - Exempelvis att kortägaren kan mata in PIN-koden direkt i terminalen.
- Analysera om hoten är rimliga.
- Analysera om protokollet hanterar dem.

Formell notation

Example (Protokollbeskrivning)

Två principals P, P' ska kommunicera.

1. P skickar sitt namn till P'.
2. P' svarar med ett token t_P för vidare användning, detta är krypterat med P:s kryptonyckel k_P.

Example (Formell beskrivning)

Principals P, P', token t_P, P:s kryptonyckel k_P.

$$P \rightarrow P' : P$$

$$P' \rightarrow P : \{t_P\}_{k_P}$$

Formell notation
Tentamen

Example (Autentisering MIUN)

Låt T vara tentanden, V tentamensvakten, n_T det unika numret för T och S skrivningen. Vidare låt k vara en kryptonyckel delad mellan legitimationsutfärdaren och tentamensvakten (legitimation).

$$V \rightarrow T : n_T$$
$$T \rightarrow V : \{T\}_k, n_T$$
$$T \rightarrow V : \{T\}_k, n_T, S$$

Protokoll och attacker

En bättre metod för fjärrlås

Example (Fjärrlås)

Låt A, B vara principals, n nonce, k_A en nyckel unik för A.

$$A \to B : A, \{A, n\}_{k_A}$$

Egenskaper

- Nonce n för färskhet.
- Krypteringen för identifiering.

Protokoll och attacker
Nyckelhantering

- Måste hantera nycklarna k_i för alla enheter i.
- *Nyckeldiversifiering*: huvudnyckel k_M och generera $k_i = \{i\}_{k_M}$.
- Måste tänka efter:
  - 128-bitar nyckel krypterar 16-bitar ID, mindre lämpligt för diversifiering.
  - Svagt chiffer ger också dåligt resultat.
  - $k_i = i \oplus k_M$?

Protokoll och attacker

Kolla nonces

Kolla nonces långt tillbaka i tiden.

- Jämför med senaste nonce.
- Spela in två och spela upp dem varannan gång.
- Förbetalda elmätare, köp två laddningar och använd dem om vartannat.

Protokoll och attacker

Betjäntattacken

- Hur genereras nonces?
- En person som har tillfällig åtkomst att generera tokens.
- Generera ett antal, använd dem senare.
- Exempelvis engångskoder för att logga in hos internetbanken.
- Attacken fungerar om nonces är (pseudo)slumptal.

Protokoll och attacker

Kontra betjäntattacken

Förbättring

- Använd en räknare c som successivt ökas på.
- $A \rightarrow B$: $A, \{A, c+1\}_{k_A}$, $c = c + 1$.
- Inget $c' \leq c$ accepteras.

Problem

- Får inte ha jämförelsen $c' = c$, ger synkroniseringsproblem.
- $c \notin \mathbb{Z}_+$ utan $c \in \mathbb{Z}_{2^x}$, för något $x \in \mathbb{N}$: vid något tillfälle blir då $c + 1 < c \pmod{2^x}$.

Protokoll och attacker
Andra tillämpningar

- Tillbehörskontroll: skrivare ändrar inställning från $1200\ \mathrm{dpi}$ till $300\ \mathrm{dpi}$ om icke-originalbläckpatroner används.
- ”Använd alltid godkända originaldelar”.
- Inte hålla angripare ute, utan hålla användare inne.
- Läs kapitel 7 *Economics* i [And08] för vidare diskussion.

Challenge–response

Grundläggande princip

Två principals A, B med gemensam nyckel k och nonce n.

$$A \to B\colon n$$
$$B \to A\colon \{B, n\}_k$$

Problem

- Dåliga (pseudo)slumptalsgeneratorer, ger förutsägbara n.

Challenge–response

Tvåfaktorautentisering

- Ha användarnamn och lösenord.
- Komplettera med extern kod; exempelvis genererad av koddosa, SMS till mobiltelefonen.
- Finns många varianter, kombinera två:
  - Något du vet (lösenord),
  - något du har (koddosa, mobiltelefon),
  - något du är (biometrik).

Challenge–response
Tvåfaktorautentisering

Protokoll (tvåfaktorautentisering med koddosa)

Låt A, B, D vara principals, D är koddosa, k är nyckel delad mellan B, D och p är A:s PIN-kod.

$$A \to B: A$$
$$B \to A: n$$
$$A \to D: n, p$$
$$D \to A: \{n\}_k$$
$$A \to B: \{n\}_k$$

Mittuniversitetet
MID SWEDEN UNIVERSITY

Challenge–response
Tvåkanalsautentisering

Protokoll (tvåkanalsautentisering med mobiltelefon)

Låt A, B, M vara principals, M är mobiltelefon och p är A:s lösenord.

$$A \to B: A, p$$
$$B \to M: n$$
$$M \to A: n$$
$$A \to B: n$$

Miljöbyte

- Betalkortsystemet designades för en pålitlig miljö.
- Kraftigt reglerad miljö inbyggd i bankens fasad.
- Tillämpas i den mindre pålitliga miljön i samtliga affärer.
- Skimming.

Miljöbyte
Personen i mitten

- ”Det är enkelt att spela oavgjort mot en schackstormästare i postschack: spela bara mot två stormästare samtidigt, en som vit och en som svart, och skicka deras brev mellan varandra.” (John Convey)
- Problem med pålitliga användargränssnitt: hur vet du att inte kortterminalen ljuger?

Internetbanken och betalkort

Olika former av bankdosor

Swedbank

- Individuell dosa, förkonfigurerad av banken.
- Kan generera engångskod.
- Kan hantera challenge–response.

Nordea

- Oberoende smartkortläsare, använder individuellt betalkort.
- Kan generera engångskod.
- Kan hantera challenge–response.

Internetbanken och betalkort

Problem som kan uppstå

Problem

- Om bankkort och dosa förvaras tillsammans kan PIN-koden utläsas från de slitna knapparna på bankdosan.
- Om kortet används i en dålig terminal har angriparna allt som behövs för att logga in till ditt bankkonto.

Förbättringar

- Använd inte samma säkerhetsmekanism i flera sammanhang.
- Ha separata oberoende mekanismer.
- Ha ett pålitligt användargränssnitt.

Overview

Trusted Platform Module

Overview

Watermarking

- A different approach has to be taken for non-executable content, since this material cannot check itself.
- The approach here is watermarking using steganographic methods.
- However, these are also quite easily thwarted.

Referenser

[And08] Ross J. Anderson. *Security Engineering. A guide to building dependable distributed systems*. 2nd ed. Indianapolis, IN: Wiley, 2008. ISBN: 978-0-470-06852-6 (hbk.) URL: http://www.cl.cam.ac.uk/~rja14/book.html.