

Brmdoor_libnfc short

čím se liší od předchozích brmdoorů

abyssal • 7.6.2018

Rozdíl od starých brmdoorů

- pěkný kód, jasný config, rozšiřitelné
- všechno zdokumentované, procesy i kód
 - <https://brmlab.cz/project/brmdoor>
- jednoduchá integrace s JendaSAPem
 - https://brmlab.cz/project/brmdoor/start#adding_a_new_card_to_jendasap_and_import_to_brmdoor
 - import z JendaSAPu přepíše UID databázi nanovo

Featury (pokračování původních)

- Více způsobů autentizace
 - podle UID karty
 - víme, že jsou čínské s R/W UID
 - Yubikey Neo challenge-response
 - Desfire s podpisem UID v NDEF zprávě
 - zatím nikdo nevyrábí Desfire s přepisovatelným UID
 - pro případ, že by někdo zkopíroval DB a vydával se za jiné členy
- IRC reporting, více channelů, OPEN switch

Hardware

- Raspberry – všechny verze (1, 2, 3)
 - distro ideálně raspbian, ale fungovali by i jiné distra, stačí mít devel verzi knihoven
- PN532 Adafruit shield
 - jsou 2 verze, ale rozdíl je jen v umístění PINů
- zámeček – používáme BERA-E (D?), ale fungoval by i jednodušší „bzučák zámeček“

Implementace

- Python + C++ handling čtečky a APDU n a čtení UID, Yubikey a NDEF via libnfc/libfreefare
- 3 separátní vlákna
 - 1 odemyká, 1 na IRC (optional), 1 na open/close switch (optional)
- drží čtečku „otevřenou“ a jenom se ptá „je karta?“ se sleepem místo posílání celé otevři/přečti/zavři u nfc-list

Implementace (2)

- SQLite databáze, lze přidávat/měnit lidi bez restartu daemona
- spouštěno automaticky přes systemd v screenu
- dobře čitelný log s vícero levely (debug je fakt verbose, hlavně kvůli IRC)
- OPEN/CLOSE switch umí reagovat na libovolný soubor, nemusí to být jen GPIO
- zapomenuté přepnutí lze změnit via /topic

Challenges

- správné použití python-irc (založeno na python-twisted, yay)
 - zjištění topicu vyžaduje state machine
 - umí to reconnect, ale handlers se ztratili (nedokumentováno)
 - mnoho různých druhů disconnectů, co se muselo vyzkoušet prakticky, simulace nebyla dostatečná
 - vyladěno

Challenges (2)

- posílání APDU (Yubikey a Desfire NDEF)
- USB a SPI komunikace PN532 mají několik rozdílů
 - USB verze umí příkaz „spi dokud se neobjeví karta v poli“
 - u SPI to způsobovalo 100% CPU – SPI nemá interrupt, i když čtečka ho má vyvedený
 - místo poll-ování se používá na SPI „přečti karty v poli a pak spi 0.2 sec“)

Co by šlo dál

- nápad vypínat zásuvky s pájkami, když se přepne OPEN/CLOSE

Technické zajímavosti

- čtečka umí přečíst několik karet v poli (experimentálně 5, standard to nelimituje, spíš výkon)
- tzv. antikolize, podobně jak u Ethernetu
- prakticky to není využito
- jen připomínka, že pokud v pěnížence máte vícero karet vedle sebe, absolutně to neznamená, že je nelze přečíst

Thanks

abyssal