PropLANE

Kind of keeping the NSA from watching you pee
Introduction

• The guys up here
  – Mark Carey (phorkus)
  – Russ Rogers (russr)
  – Ryan Clarke (LOstboy)
  – Rob Bathurst (evilrob)

• Guys not up here
  – You
History of Crypto Part I

- Scytale
- Caesar Cipher
- One Time Pad (OTP)
- Enigma Machine
- SIGABA
- Data Encryption Standard (DES)
- Advanced Encryption Standard (AES)
Recent Things in History

• The NSA vacuum
• Is TOR safe!?!?!
• The Freenet Project
The Project
The Pile
The PropLANE

• The Idea!
  – .gov style network protection for the masses

• Why did we do this?
  – we too like to keep our shit, our shit, and just our shit

• How did we do this?
  – DARPA CFT
The Parts Part I

• DC 20 Badge
  – Parallax Propeller Chip
  – 16 User I/O Pins
  – SPI Boot ROM
  – TTL Serial-to-USB
  – Infrared Transceiver
The Parts Part II

• Additional Items
  – Ethernet Transceiver
    • Microchip ENC28J60
    • 3.3/5v
    • 8k Static Ram Buffer
    • If you don’t use this, you will have to write your own driver
  – SD Card (keystore)
    • Almost any SD card will work
The Software

• Spin
  – “high level” programming language
  – byte code interpreter
  – learn.parallax.com

• PASM
  – Propeller Assembly
  – Faster
  – pPropellerSim/GEAR
Fair Warning

• Synthesized SPI using specialized COG instructions
• Transparent bridging
• Small key size (128 vs 256) due to size constraints
Warning About Crypto

• Why crypto works
  – Hash vs Encryption

• Crypto can be defeated
  – Losing your symmetric key
  – Compromised PKI
  – Brute Force
  – Poor Implementation
Money Shot
The Problem
Approach

• Cheap
  – Propeller
  – Arm

• Fast-ish
  – Propeller (not so fast)
  – ARM (can be fast)
  – FPGA (screaming fast)

• Easy to use
  – Simple key exchange
  – ON/OFF switch
The Badge
DC 20 Badge
Badge Schematic
Pinout
Pinout
PropLANE Software
How a Propeller Works

• Cogs
• Jobs
• Spin/PASM
• What if I want to port it?
The Crypto Cog

- Encrypt Cog
- Decrypt Cog
- Speed Test
- Basic Sequence
  - Packet In
  - Mem Copy
  - Decrypt
  - Read/Write
  - Encrypt
  - Mem Copy
  - Packet Out
The Network Cog

• Network Comms Design
  – 2 SPI Cogs
  – “Big Shovels”
  – Packet Queue

• Packet Wrapping
  – Payload Encryption
  – Convert to Proto 99
  – TCP/UDP signal bit

• Targeting
  – key to network relationship
Key Management

• Multi-Key management is a joy
• Suggested Protections
  – Encrypt keys for the destination device
  – Never transmit in plaintext
  – Use alternate channels if possible
• Separated communications channel
  – SD Card/IR
Using the PropLANE
Badge Assembly
The Basics

• How to enroll your friends
  – Key.txt

• Protections the PropLANE provides
  – Encrypts communications on the blackside

• What the PropLANE won’t do
  – Fancy shit

• What you shouldn’t use the PropLANE for
  – Hiding from the Government
  – Banking
  – The lulz
Danger Will Robinson

• Crypto Implementation
  – Key size limitation
  – Speed
  – Single Key per device
    • It does not have to stay this way

• Expected privacy
  – If the key is not compromised, you’re doing pretty good

• Difficulty in creating the PropLANE
  – Lots of beer, long nights, and pain
Future Goals

• Where we’d like to take the project
  – Try new algorithms (SIMON, SPECK, EU)
  – Complete a ARM port
  – Any direction you want

• What we think we can do in the future
  – Make crypto a feature on future electronic DC badges
  – Help protect the community and give people something to hack on
Administratum

• Where can I get the software and instructions?
  – https://github.com/proplane/proplane

• Where can I find more information?
  – http://www.proplane.org

• Contact info
  – firstname@proplane.org

• Drink Preference
  – Any
Questions?