Nindendo DS

Introduction and Hacking

marcel@koeln.ccc.de, mm, tobias@koeln.ccc.de

Chaos Computer Club Cologne e.V.
http://koeln.ccc.de

29.12.2006

Outline

Outline

Introduction

"'“Nintendo sold 26.82 million units of the Nintendo DS"'
(N-Sider.com, 01.12.2006)

Hardware

- 2 TFT LCD screens, each 256x192 pixels resolution, lower one touch-sensitive
- Main CPU: ARM946E-S (66 MHz)
- Co-processor: ARM7TDMI (33 MHz) (GBA)
- 4MB main memory
- Wi-Fi 802.11b, simple proprietary "‘NiFi"’ protocol for local games, standard TCP/IP for internet play
- A/B/X/Y/L/R Buttons, Start, Select, digital control pad
- integrated microphone and stereo speakers
- Up to 10 hours of battery life (proprietary Li-Ion battery pack)
- GBA slot, NDS slot (Slot-1)

GBA

- Main CPU: ARM7
- Co-processor: Z80 (Gameboy)
- Usual stuff . . .
- No copyprotection

Gameboy Timeline

1989, 1998

2001, 2003, 2005

Nintendo Firmware

- includes basic GUI for user settings and PictoChat
- Communication between DS and cartridges in Slot-1 is encrypted, only game header is read unencrypted.
- Downloaded games need to be RSA signed
- flashable by NDS code only

Jumper

The first area of the firmware is read-only. A jumper needs to be shortened to gain write access.

Outline

GBA Flash Cards

- There are several ways to enable the first homebrewn code execution.
- Many people still have their GBA flash cards.
  - ➔ use the GBA slot to run NDS code from an SD card.

EZ IV Lite

G6 Flash

M3 SD Slim

Problem: How to run DS code from the GBA slot?

Subverting Download Play: WiFiMe

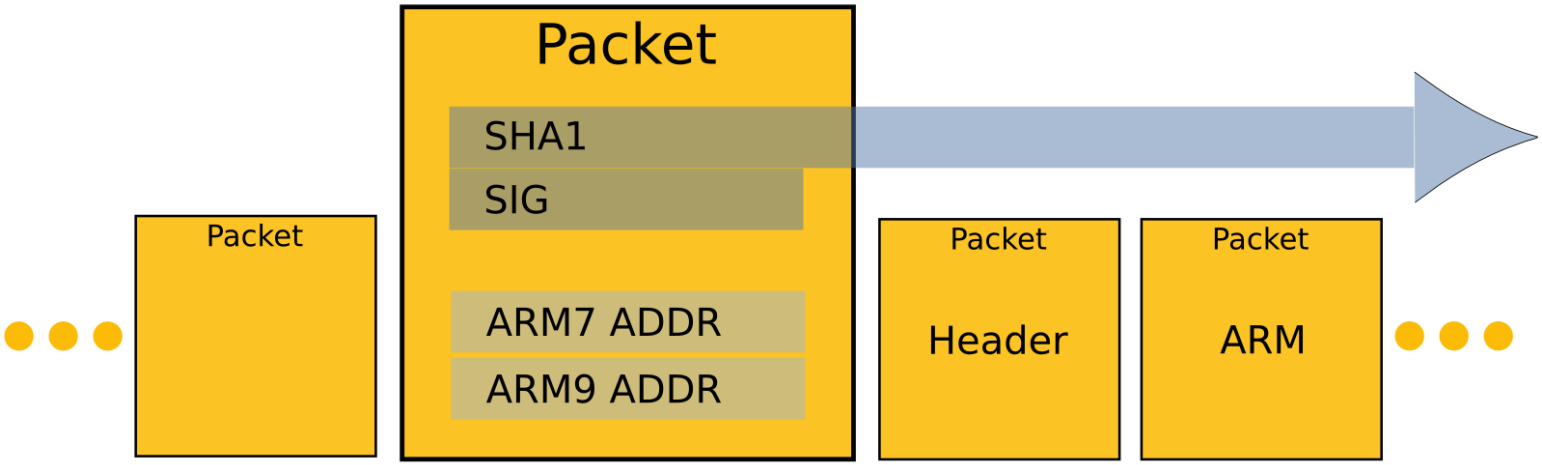
Uses a conceptual error in the download play code.
When sending the code for download play the hosting DS sends a RSA frame, which includes:

- Signature block for the following NDS header, ARM7 and ARM9 code.
- ARM9 execute address
- ARM7 execute address
- and some other uninteresting stuff...

The RSA frame itself is not signed!

WiFiMe

Replaying a complete game download, only exchanging the ARM7 and ARM9 execute address.



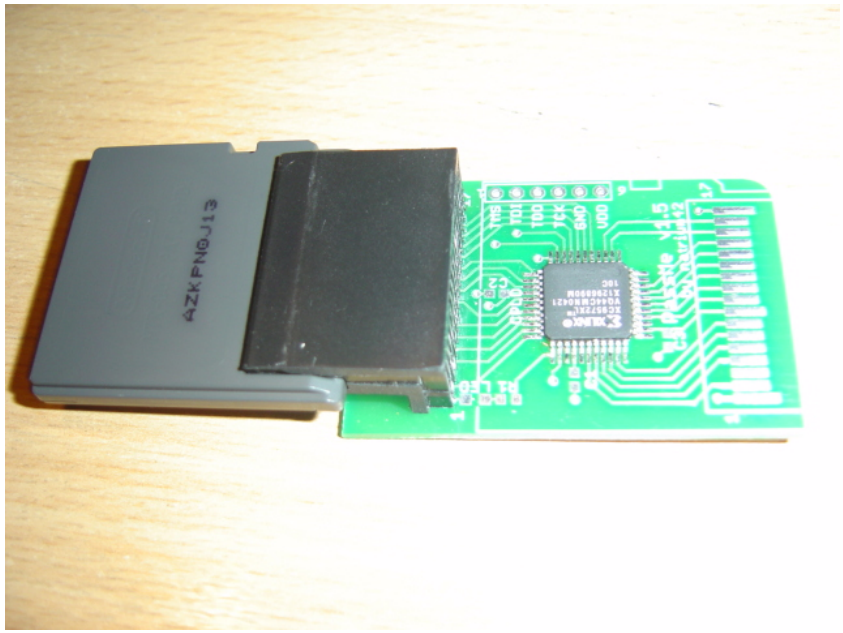
Subverting Game Copy Protection: PassMe

- Patches original ROM header to start from GBA slot
- It's possible to run homebrew code
- Hardware solution, tricks DS into authenticating a commercial game

Passme

Nintendo Fixes Bugs

Newer versions of the DS are ‘fixed’ to defeat WiFiMe and PassMe. Download play does not trust the execution address in the RSA anymore. Firmware does not let cartridges specify a start address in the GBA address space.
The homebrew community reacts: PassMe2. . .



PassMe2

PassMe2

- Update of PassMe, works with newer firmware versions
- Uses evil tricks to achieve the same goal as PassMe
- Since it can not let the firmware jump directly to the GBA address space, it lets the firmware jump somewhere into the game binary which let the ARM7 jump into GBA SRAM address space
- Since the instructions are at different places in different games, it needs to be programmed for the game it will be used with
- More evil tricks get us out of GBA SRAM and execute our code from the GBA card.

Copy Protection Cracked: Slot-1 Flash Cards / NoPass

NoPass is a equivalent to a PassMe, but it needs no game and fits completly into the NDS slot.
Newer flash cards use the NDS slot to run NDS code, possible because encryption was reverse engineered.

Ninja DS / DS-Xtreme / Magic Key / M3 Simply

Outline

FlashMe

FlashMe was developed by "‘Loopy, FireFly und DarkFader"’. It allows to run NDS code from the GBA slot.

- Homebrew software
- "‘Pirated"’ game
- WiFi games work
- restore function (Un-Bricker) fits into write protected firmware space
- no more RSA signature check on download play
- no health warning screen anymore (optional)

Homebrew

- DSOrganize
- Moonshell
- WiFi-lib
- ...

Outline

Linux On Every Device!

DSLinux is a Linux port for the Nintendo DS mainly developed by Malcolm ‘pepsiman’ Parsons, Stefan ‘stsp’ Sperling and Wolfgang ‘Amadeus’ Muess.

- The DS has no MMU, so DSLinux is based on uClinux
- Preferred port now uses the RAM present on the SuperCard/M3, so DSLinux has 36MB of RAM available.
- Thus currently support for Slot-1 flash cards is not planned
- Supports all DS hardware (except sound input)
- Currently no GUI

Linux Applications

- busybox
- dropbear SSH
- links with SSL support
- bitchX IRC client
- madplay
- bsdgames

Outline

Emulators

Emulating The NDS

- Dualis - DS emulator for windows, emulates only demos and no NDS dumps
- iDeaS - DS emulator for windows, emulates demos and NDS dumps
- DeSmuME - OpenSource DS emulator, runs demos and NDS dumps, also available for linux
- NO$GBA - small and fast GBA/NDS emulator for DOS/Windows

Emulators

NDS Emulating Other Devices

- SnezziDS - SNES emulator
- snesDS - SNES emulator
- PocketSPC - SNES sound chip emulator
- nesDS - NES emulator
- ScummVM DS - ScummVM Port
- CalcEmu - emulates the TI 85 calculator

Outline

WFC

Nintendo WiFi Connection

Multiplayer uses an ad-hoc W-LAN or Nintendo's WFC via the internet.

- some games use gamespy master servers
- P2P games like Mario Kart
- Authentication with the WFC uses SSLv3 (+ server certificate check)
  NDS ID + Game ID ?
- Authentication with WFC enables the NDS IP to use T-Online hotspots in Germany without a fee!
- Hey, look! The URL in the ROM is in plaintext...

Games

> *"'Not to mention that there are actual games out for it."' (slashdot.org)*

Actually, DS games are pretty fun!

Gamez

December 2006:

Europe	228 titles
USA	312 titles
Japan	443 titles

Future

- reliable Slot-1 flash cards.
- Linux GUI and applications
- more WiFi homebrew
- DS-PSP WiFi multiplayer?
- . . .

Literature and Links

- http://wiki.pocketheaven.com/
- http://masscat.afraid.org/ninds/wifi_investigation.php
- http://akkit.org/dswifi/
- http://darkfader.net/ds/
- http://devkitpro.org/
- http://www.bottledlight.com/ds/ (NDS Tech Wiki)