

PA193 - Secure coding principles and practices



Security Code Review

Petr Švenda svenda@fi.muni.cz

CRCS

Centre for Research on
Cryptography and Security

www.fi.muni.cz/crocs

Security code review – simple.zip

- Download simple.zip from IS
 - Try to build and run it
- Perform security code review (bottom-up approach)
 - Run static / dynamic analysis tools (more of them)
 - Inspect source code and make high-level overview
 - Make call graph (by hand or automatically)
 - Find and document problems

Problem identification: DSA-1571-1 openssl

Severity: critical

Risk: high - directly exploitable by external attacker

Problem description: crypto/rand/md_rand.c:276 & 473 – The random number generator in Debian's openssl package is predictable...

Remediation: revert back to usage of uninitialized buffer *buff*

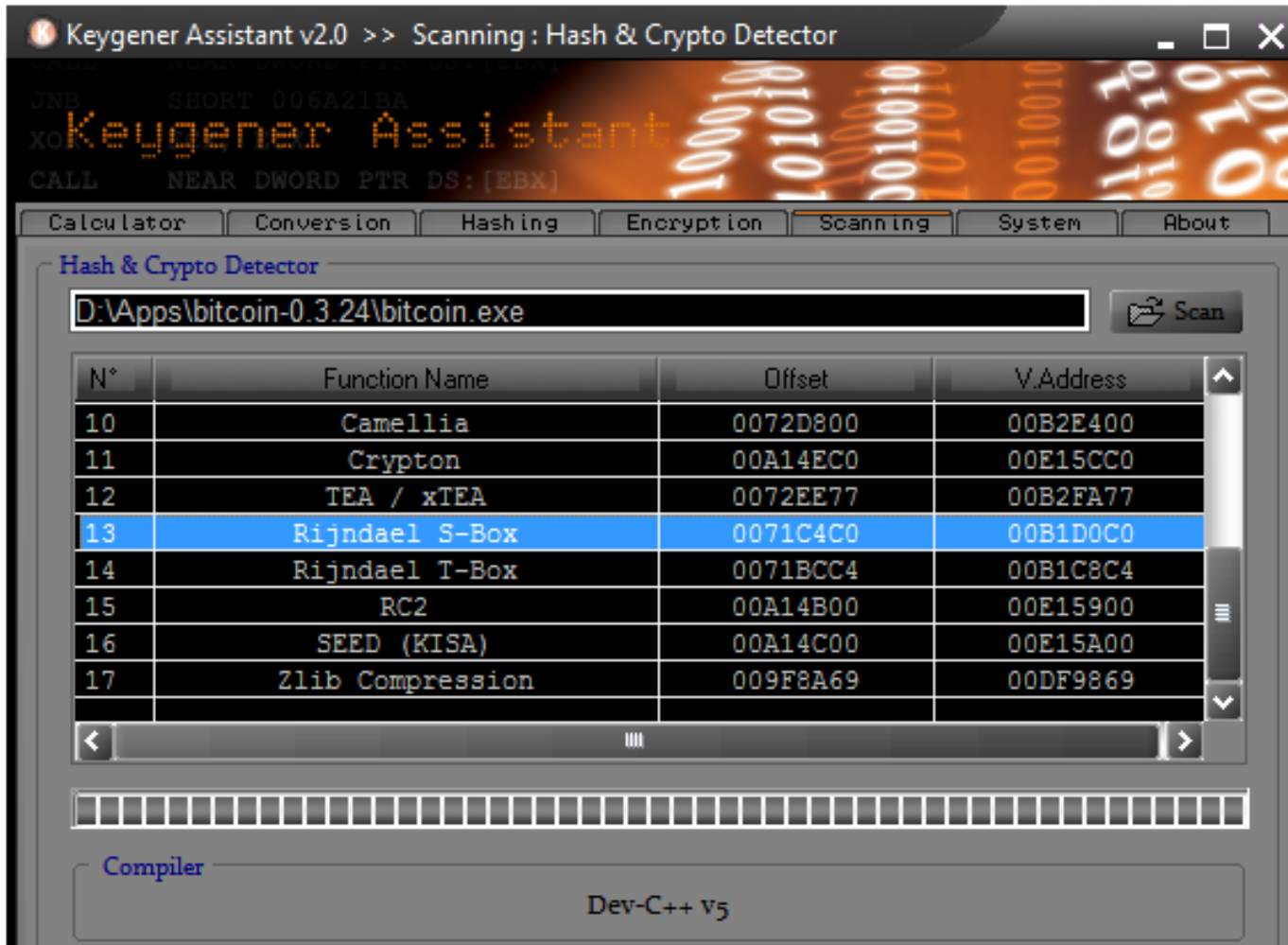
Some hints

- Classes of problems
 - Bad usage of cryptography
 - Keys
 - Unsafe functions
 - Race condition
 - Memory corruptions
 - Information leakage
 - Sensitive data handling
- *(Don't focus on Blowfish.cpp – no intentional issues there)*

Optional: Security code review – large one

- Download larger security project by your selection
 - Wincp, GnuPG, KeePass, Putty, Bitcoin...
- Inspect compiled binary
 - Try to detect what crypto algorithms were used
 - <http://at4re.com/download.php?view.30>
- Get idea about structure of code
 - Can you find where user password is supplied and processed?

Usage of crypto somehow hidden?



The screenshot shows the Keygener Assistant v2.0 interface. The title bar reads "Keygener Assistant v2.0 >> Scanning : Hash & Crypto Detector". The main window has a dark background with a binary code pattern. The text "Keygener Assistant" is displayed in a large, stylized font. Below the title bar, there are tabs for "Calculator", "Conversion", "Hashing", "Encryption", "Scanning", "System", and "About". The "Scanning" tab is active, and the "Hash & Crypto Detector" section is visible. A text box contains the file path "D:\Apps\bitcoin-0.3.24\bitcoin.exe" and a "Scan" button. Below this, a table lists detected cryptographic functions:

N°	Function Name	Offset	V.Address
10	Camellia	0072D800	00B2E400
11	Crypton	00A14EC0	00E15CC0
12	TEA / xTEA	0072EE77	00B2FA77
13	Rijndael S-Box	0071C4C0	00B1D0C0
14	Rijndael T-Box	0071BCC4	00B1C8C4
15	RC2	00A14B00	00E15900
16	SEED (KISA)	00A14C00	00E15A00
17	Zlib Compression	009F8A69	00DF9869

At the bottom of the window, there is a "Compiler" section with the text "Dev-C++ v5".