aursec - A blockchain approach to securing software packages

Lukas Krismer & Bennett Piater

November 2, 2016

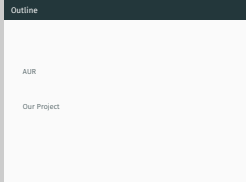
Universität Innsbruck - QE - Christian Sillaber

2016-11-02

aursec - A blockchain approach to securing software packages

Outline

1 min L



AUR

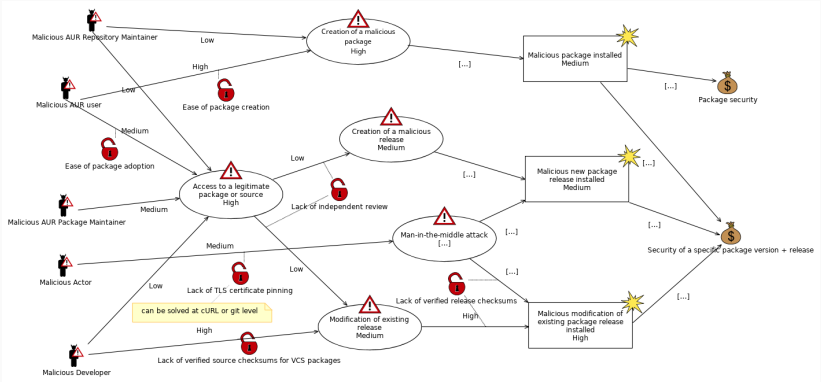
AUR

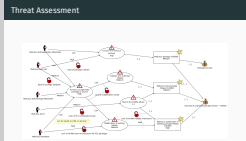
- **AUR**=Arch Linux User Repository
- Contains package build scripts (PKGBUILDs)
- Packages can be voted for inclusion in the official repositories
- Easy to use using so-called AUR helpers
- Everybody can upload PKGBUILDs
- Anyone can adopt orphaned packages

AUR
- AUR=Arch Linux User Repository
- Contains package build scripts (PKGBUILDs)
- Packages can be voted for inclusion in the official repositories
- Easy to use using so-called AUR helpers
- Everybody can upload PKGBUILDs
- Anyone can adopt orphaned packages

2min L

Threat Assessment



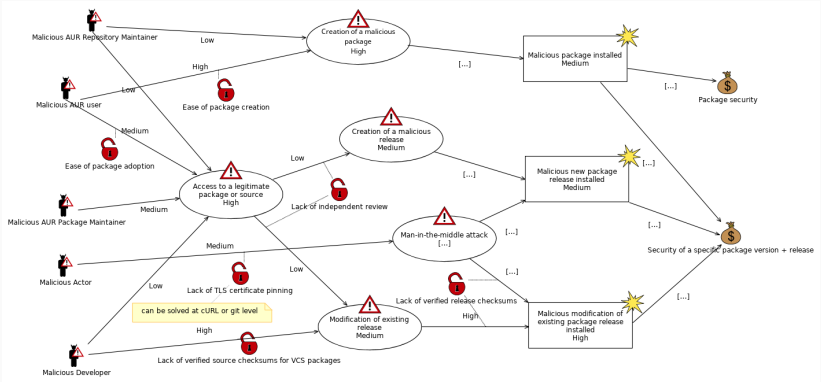


2 min B | Besonderes Augenmerk auf:

- Die grundlegenden Probleme der AUR sind praktisch unlösbar
- Zu viele haben Zugang zu Quellen und/oder Buildskripten
- Daher: Server-Seitige Signaturen würden nur MITM verhindern
- Bösartige Pakete, Releases oder Veränderungen sehr einfach

Our Project

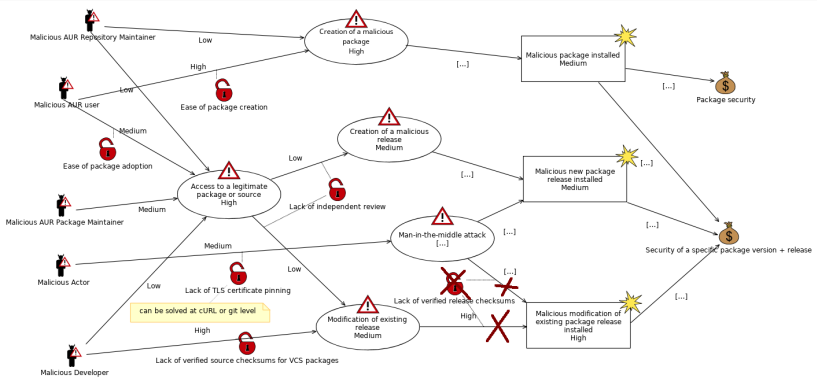
Covered Threats



Covered Threats

1 min L

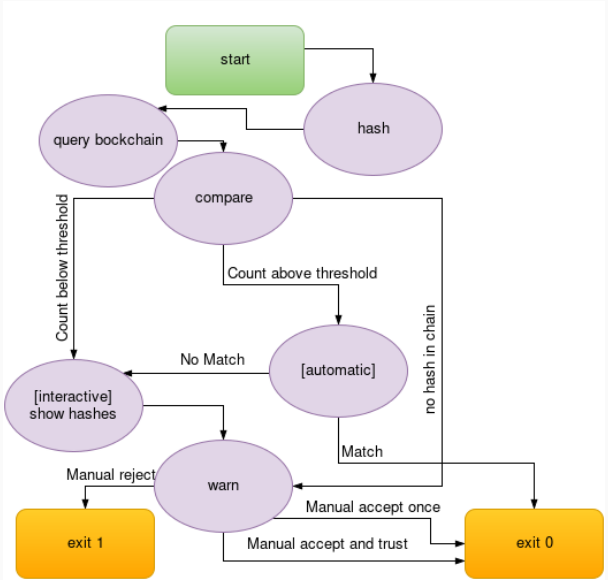
Covered Threats

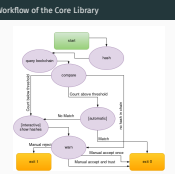


Covered Threats

1 min L

Basic Workflow of the Core Library





3 min L

Components

- Program on a private Ethereum blockchain
- Shell library
- AUR package
- Integration in aurutils
- Threat analysis of the AUR and our software
- Web- and/or CLI-Interface for stats/events

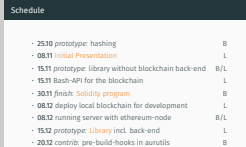
Components
- Program on a private Ethereum blockchain
- Shell library
- AUR package
- Integration in aurutils
- Threat analysis of the AUR and our software
- Web- and/or CLI-Interface for stats/events

2 min B

- Das eigentliche Programm zum Speichern der Hashes
- Unsere Library, die den Workflow automatisiert
- Ein Paket für die AUR
- Integration in einen der Besten AUR-Helper
  –> Im Zuge dessen allgemein nützliche Beiträge dazu
- Threat-analysen, um die Gefährdungsstufe und die Qualität unseres Beitrags einzuschätzen
- Ein Interface, mit dem die Aktivität der Blockchain überwacht werden kann

Schedule

- **25.10** *prototype:* hashing B
- **08.11** Initial Presentation L
- **15.11** *prototype:* library without blockchain back-end B/L
- **15.11** Bash-API for the blockchain L
- **30.11** *finish:* Solidity program B
- **08.12** deploy local blockchain for development L
- **08.12** running server with ethereum-node B/L
- **15.12** *prototype:* Library incl. back-end L
- **20.12** *contrib:* pre-build-hooks in aurutils B



2 min B

Wir haben eine sehr **detaillierte Planung** ausgearbeitet. Einerseits benötigen wir sie, um effizient **kooperieren** zu können und zügig voran zu kommen; Andererseits soll sie uns auch ein Maximaltempo vergeben, denn wir tendieren beide eher dazu, uns zu **überarbeiten**.

- Solidity-program auf Blockchain
- Library-Prototyp
- Beiträge zum AUR-Helper aurutils über Weihnachten

Schedule

- **10.01** *contrib:* TLS-public-key-pinning in aurutils B
- **10.01** configuration and trust-cutoff L
- **15.01** *test:* Integration in aurutils B
- **15.02** AUR package incl. private blockchain B
- **01.03** *finish:* libary and aurutils-Hook B
- **31.03** *finish:* Web- and/or CLI-Interface L
- **21.04** Draft paper for feedback
- **??.05** *finish:* Paper
- **??.05** Final presentation L

Schedule
- **10.01** *contrib:* TLS-public-key-pinning in aurutils B
- **10.01** configuration and trust-cutoff L
- **15.01** *test:* Integration in aurutils B
- **15.02** AUR package incl. private blockchain B
- **01.03** *finish:* libary and aurutils-Hook B
- **31.03** *finish:* Web- and/or CLI-Interface L
- **21.04** Draft paper for feedback
- **??.05** *finish:* Paper
- **??.05** Final presentation L

2 min B

- am 15.01 mit aurutils testbar
- AUR-Paket zur einfachen Verbreitung
- Programmierung endet am 31. März
- Meiste Schreibarbeit im April und besonders über Ostern
- Abgabe bequem for den Klausuren