Securing the software supply chain
About us

Jeff Hudesman – CISO at Pinwheel
• Leads the Information Security team responsible for securing and protecting information and systems while ensuring overall cyber resiliency
• Previously held global leadership positions at DailyPay, Sony, and PR Newswire
• Jeff is a graduate from the University at Buffalo with a degree in Informatics.

Peter Morgan – Co-founder/President at Phylum
• Security researcher for 20 years focused on product and application security, hardware and software reverse engineering, and software development
• Successfully led security teams at Matasano Security, Accuvant/Optiv R+D, and Clever Security
• Found and helped fix critical issues in pacemaker and implantable cardio defibrillators for two of the top medical device manufacturers
Supply chain security has hit the news cycle

- PyPI and npm Flooded with over 5,000 Dependency Confusion Copycats
  - Mar 3, 2021
  - Dec 16, 2020
- What You Need to Know about the Codecov Incident: A Supply Chain Attack Gone Undetected for 2 Months
  - Apr 19, 2021
- Hackers inject exposed Twilio SDK with malicious code
  - Jul 24, 2020
- Node.js Event-Stream Hack Exposes Supply Chain Security Risks
  - Nov 27, 2018
SCA convinced us that vulns are the only concern.

CISOs are demanding the new breed of defenses against software supply chain attacks.
Packages developed by random internet strangers
This changed out from under us

30% Code written by the company

70% Average open-source code in any given project

Package count in NPM (JavaScript)
Even the ideal isn’t great

CASE STUDY

- React
- Javascript package for website design
- Thousand of direct authors
- Nearly 7,000 possible dependencies

Tens of thousands of unknown authors could be adding code that runs on developer machines and in build pipelines
Open-source risk is more than just software vulnerabilities

- **Malicious Code**
  - Malware
  - Backdoors
  - Hard disk wipers

- **Author Risk & Reputation**
  - Package quality vs. quantity
  - Malware additions

- **License Misuse**
  - Complicated interactions
  - Corporate acquisition
  - Policed by non-profits

- **Software Vulnerabilities**
  - Cross site scripting
  - Remote code execution

- **Technical Debt**
  - Abandoned packages
  - Removed packages
  - Untested packages
How to properly think about open source risk

- **Malicious Code**
  - Malware
  - Backdoors
  - Hard disk wipers

- **Author Risk & Reputation**
  - Package quality vs. quantity
  - Malware additions

- **License Misuse**
  - Complicated interactions
  - Corporate acquisition
  - Policed by non-profits

- **Technical Debt**
  - Abandoned packages
  - Removed packages
  - Untested packages

- **Software Vulnerabilities**
  - Cross site scripting
  - Remote code execution
We have all the advantages we need

- Source code
- Revision history
- Author relationships and behaviors

- Package metadata
- Package relationships
Supply chain risks are almost always multi-faceted. Need to analyze numerous indicators to identify risks and issues.
Digging into a risk:
Typosquatting
Security automation gives us a chance
Thank you!

Q&A