FINOS

Fintech
Open Source
Foundation

FINSEMBLE

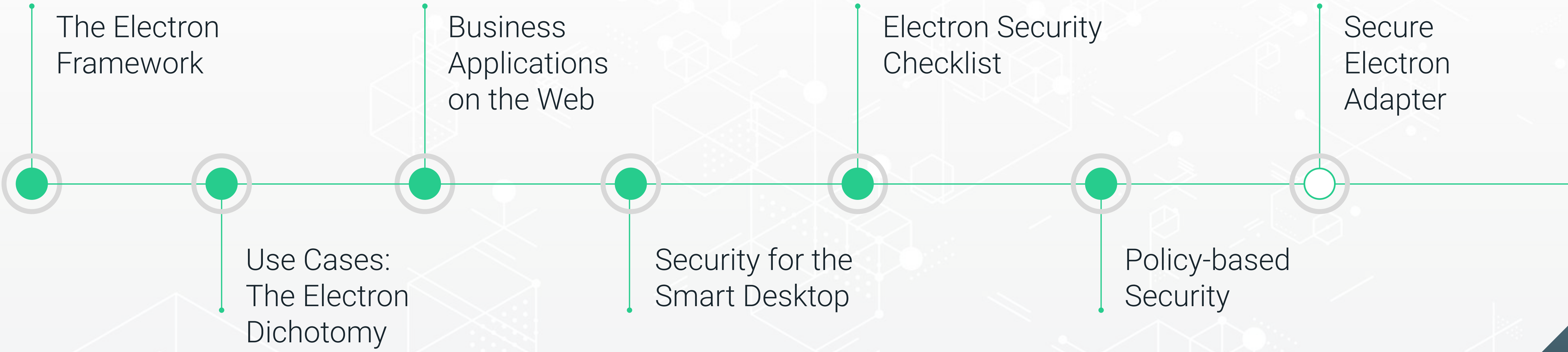
ChartIQ

# How to Secure the Electron Container for Capital Markets

FINOS Host
James McLeod, FINOS Director of Community

ChartIQ Presenters
Kris West, Director Solutions Engineering
Ian Mesner, Chief Architect

May 2020



- The Electron Framework
- Use Cases: The Electron Dichotomy
- Business Applications on the Web
- Security for the Smart Desktop
- Electron Security Checklist
- Policy-based Security
- Secure Electron Adapter

FINSEMBLE
Evernote
SYMPHONY
slack

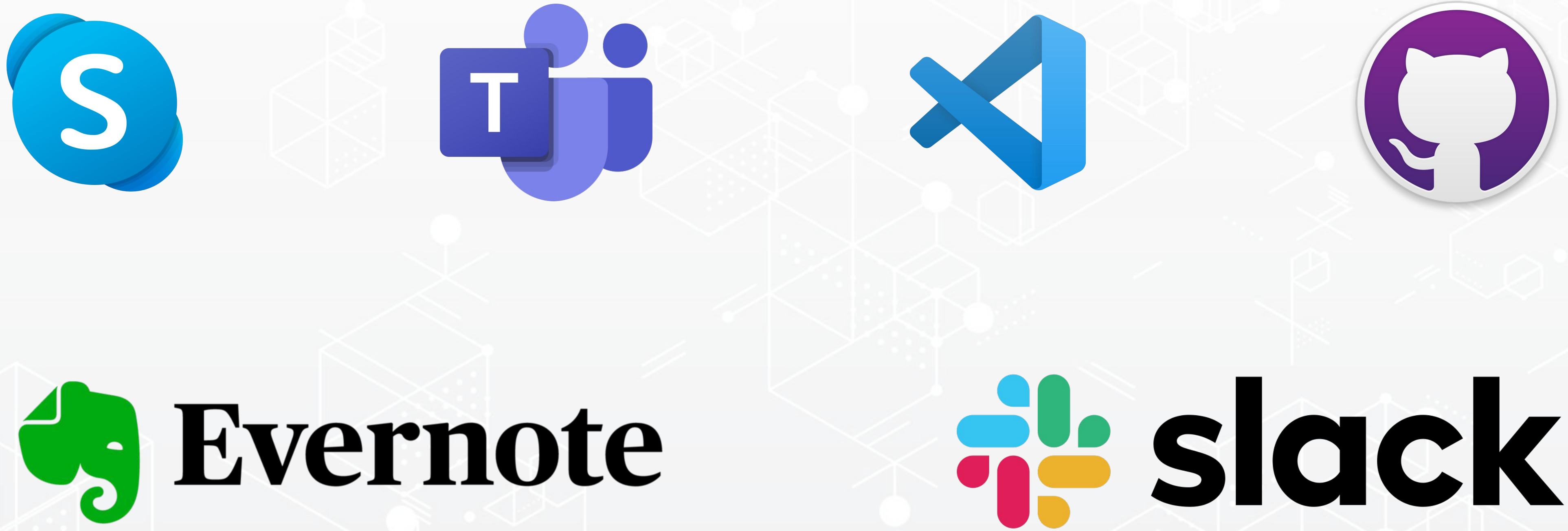
node
JS ®

+

chromium

# The Electron Dichotomy

# Do we need to secure Electron?



# Business Applications on the Web

Web developers know what they're doing:

- SQL Injection
- Cross-site scripting
- etc.

Common vulnerabilities and best practices analyzed and published by organisations:

- OWASP (top 10)
- Carnegie Mellon Uni. SEI's CERT program
- US Department of Homeland Security's
  Cyber & Infrastructure Security Agency

# Web Application Security

| | | |
|---|---|---|
| 1 | **Injection** | Buffer overflow, SQL injection; parameters |
| 2 | **Broken Authentication** | Credential theft through snooping or brute force |
| 3 | **Sensitive Data Exposure** | Storing data without proper safeguards |
| 4 | **XML External Entities** | Remote code execution of remote xml resources |
| 5 | **Broken Access Control** | Gaining access to restricted systems. |
| 6 | **Security Misconfiguration** | Insufficient access control |
| 7 | **Cross Site Scripting** | Remote code execution due to code as data |
| 8 | **Insecure Deserialization** | Data retrieval as a point of attack or remote code execution, etc. |
| 9 | **Using components with known vulnerabilities** | Failure to audit dependencies |
| 10 | **Insufficient Logging and Monitoring** | Failing to audit access. Extraneous functionality exploit of logging sensitive info |

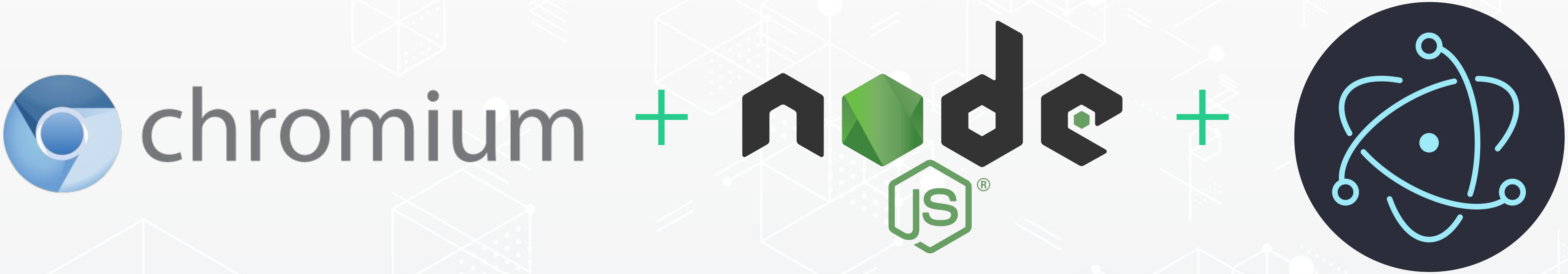
*https://owasp.org/www-project-top-ten/*

# Web Application Security: the Comfort of the Sandbox.

Browsers are designed to execute remote, untrusted code.

- Restricted operating system APIs
- Integrated sandbox
- Site isolation
- Web security policies

# But isn't Electron based on a web browser?



# Security for the Smart Desktop

**This new class of software introduces new risks to manage.**

## Integrating applications from multiple sources

Bringing a variety of technologies from a variety of software providers onto your desktop can be risky if not well managed.

## Web technology without the browser

The arrival of web technology on the desktop, outside of the browser, compounds the already complex challenge of desktop security.

## Communications and Interop

The goal of the smart desktop is to promoted communication and interoperability between applications and micro-frontends, but without compromising security.

Apple Inc. (NASDAQ:AAPL) Shares Acquired by Wealthsource Partners LLC

Princeton Portfolio Strategies Group LLC Has $12.09 Million Stock Holdings in Apple Inc. (NASDAQ:AAPL)

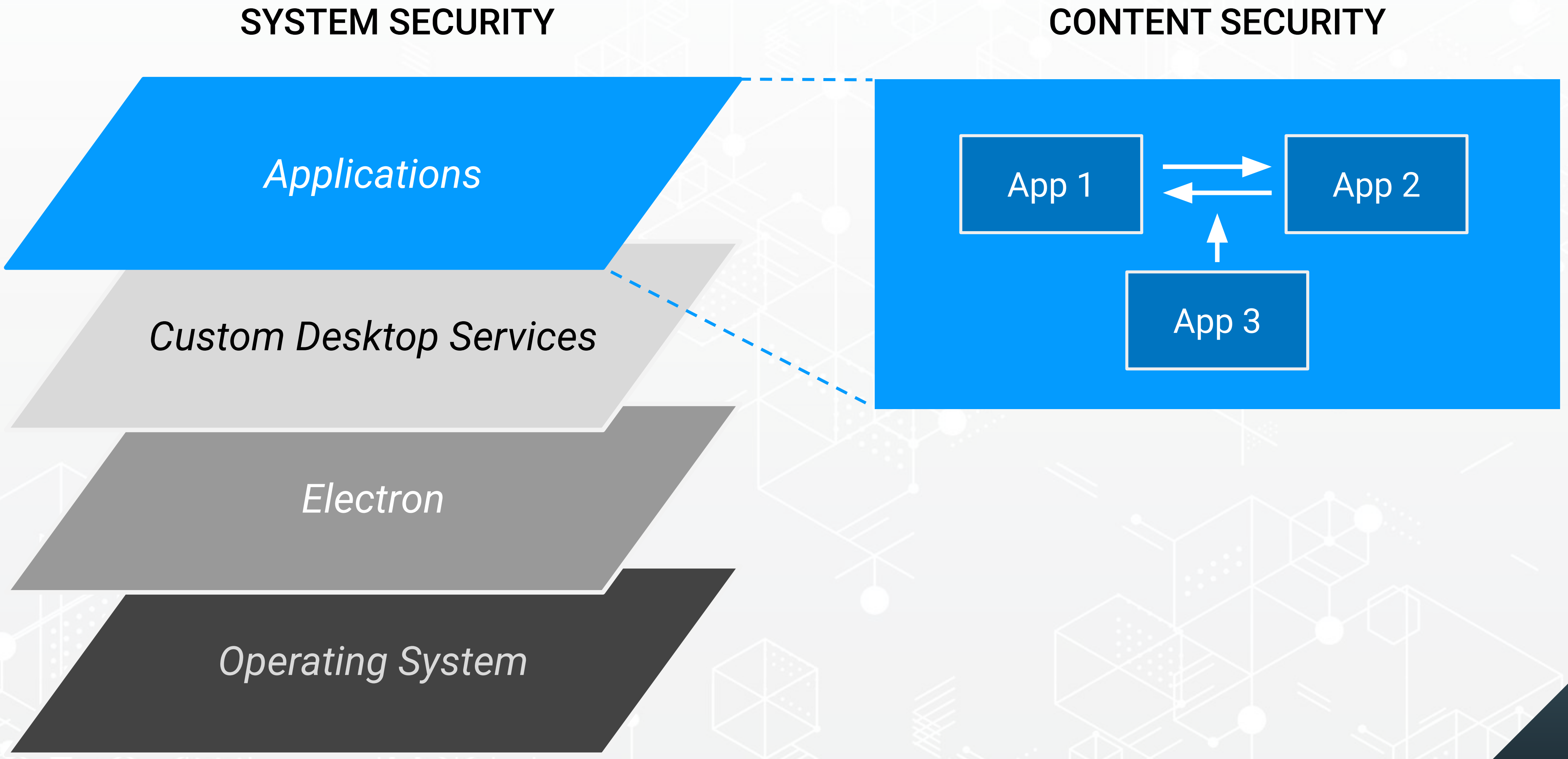
Apple Inc. (NASDAQ:AAPL) Shares Sold by Rockefeller Capital Management L.P.

Apple Inc. (NASDAQ:AAPL) Shares Acquired by Wealthsource Partners LLC

What's Next for Apple (AAPL) Stock: Holiday Shopping, iPhone 11, Apple TV+

| | |
|---|---|
| Price/Earnings Ratio | null |
| Price/Book Ratio | 7.68 |
| Sales Or Revenue | 21,461,268,000 |
| Cost Of Goods Sold | 17,419,247,000 |
| Gross Income | 4,042,021,000 |
| Selling, General, Administrative Expenses | 4,294,861,000 |

# Security for the Smart Desktop



# Security for the Smart Desktop

## System Security

| | |
|---|---|
| **Input Validation** | Buffer overflow, SQL injects, params, XSS, deserialization |
| **Broken Authentication** | Credential theft through snooping or brute force |
| **Broken Authorization** | Gaining access to restricted systems or elevated rights |
| **Dependency exploits** | Failure to audit dependencies |

## Content Security

| | |
|---|---|
| **Communication** | Ability to listen to messages meant for others. |
| **Storage** | Unauthorized access to persisted data |
| **Runtime** | Information about what other applications are running and the current user or platform |
| **Configuration** | Gain information about or modify access to runtime through config |

# Electron Security Checklist

1. Only load secure content (https, wss, sftp)
2. Disable the *Node.js integration* in all renderers
3. Enable *context isolation* in all renderers
4. Use *session.setPermissionRequestHandler*() to control what desktop API permissions remote content has access to
5. Do not disable *webSecurity*
6. Define a *Content-Security-Policy*
7. Do not set *allowRunningInsecureContent* to true
8. Do not enable *experimental features*
9. Do not use *enableBlinkFeatures*
10. <webview>: Do not use *allowpopups*
11. <webview>: Verify options and params
12. Disable or limit *navigation*
13. Disable or limit *creation of new windows*
14. Do not use *openExternal* with untrusted content
15. Disable the *remote module*
16. Filter the *remote module* (if you can't disable it)
17. Use a *current version* of Electron

https://www.electronjs.org/docs/tutorial/security

# Implement the Checklist

Content Security → ~~Interoperability~~

System security → ~~Desktop APIs~~

# Back to the drawing board...

Handle secure, trusted code differently than content from untrusted sources

### Desktop Services

- Build microservices for the desktop
- Implement interprocess comms

### Policy-based Security

- Enable/Disable Electron APIs via config
- Principle of least privilege (POLP)

# Announcing the Secure Electron Adapter

At ChartIQ, we believe in both:

- Open Source software
- Collaboration

Secure Electron Adaptor (SEA)

- Adheres to Electron's own security recommendations by design.
- Provides support for policy-based security, making it much easier to work with
- Implements inter-process communication, filtered by  that policy-based security

SECURE
ELECTRON ADAPTER

FINOS

IQ

# Secure Electron Adapter Next Steps

**Where can I get it?**

github.com/finos/secure-electron-adapter

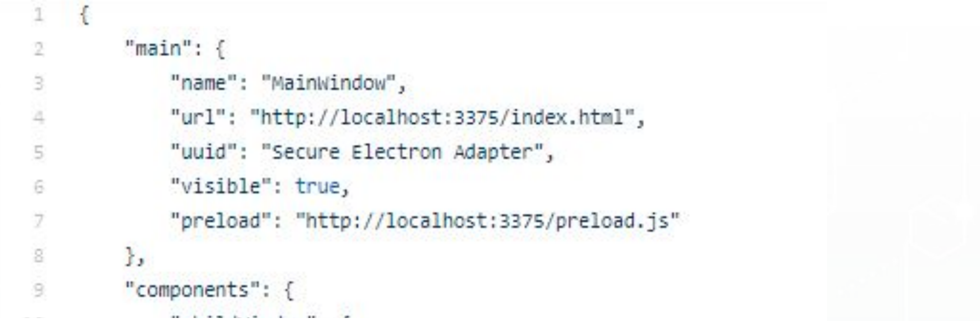
**Quick-start project**

github.com/finos/sea-quick-start

- Minimal Electron app using SEA
- Based on the Electron quick start guide
  https://www.electronjs.org/docs/tutorial/quick-start

SECURE
ELECTRON ADAPTER

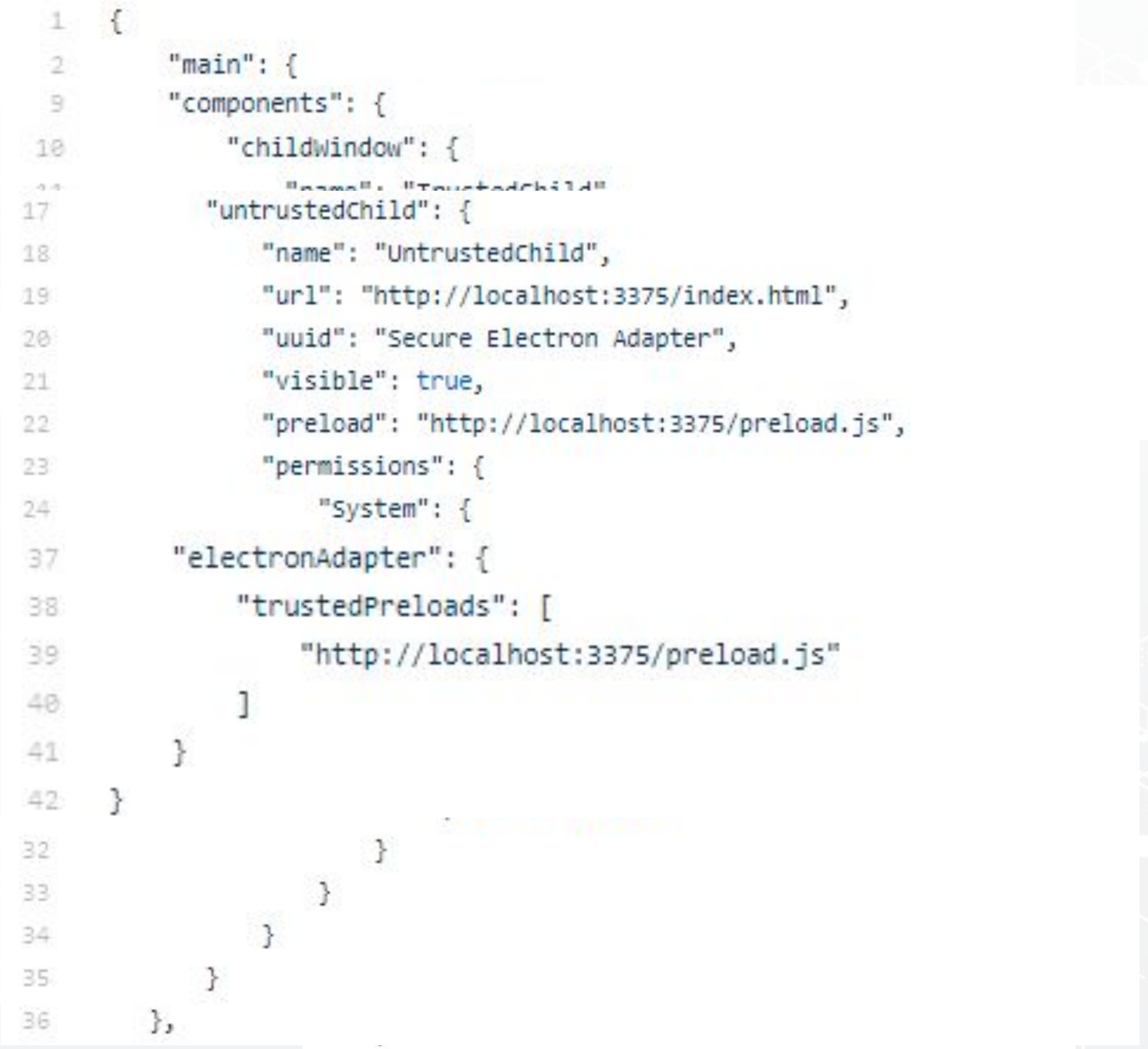
# Secure Electron Adapter



```
{
    "main": {
        "name": "MainWindow",
        "url": "http://localhost:3375/index.html",
        "uuid": "Secure Electron Adapter",
        "visible": true,
        "preload": "http://localhost:3375/preload.js"
    },
    "components": {
```

## SEA is config-driven

- **/public/manifest-local.json**
  Used to configure:
  - Main process
    - Loaded from a remote location
    - Can be a visible window or service
    - Can have content preloaded into it
  - Other 'components'
    - Also loaded from a remote location
    - Can have permissions specified
  - Electron adapter settings
    - Such as 'trusted' preloads
      (ideal for creating clients for your own desktop services)



```
{
    "main": {
    "components": {
        "childWindow": {
            "name": "TrustedChild"
      "untrustedChild": {
          "name": "UntrustedChild",
          "url": "http://localhost:3375/index.html",
          "uuid": "Secure Electron Adapter",
          "visible": true,
          "preload": "http://localhost:3375/preload.js",
          "permissions": {
              "System": {
    "electronAdapter": {
        "trustedPreloads": [
            "http://localhost:3375/preload.js"
        ]
    }
}
                    }
                }
            }
        }
    },
```

## SEA Quick Start Demo

# What does SEA not do?

SEA doesn't include:

- Detailed desktop services
- Ready-made UI components
- Full solutions to the 'Big 8'
  (Rather its focused on secure foundation on which to build these)
- Support services

Need more?

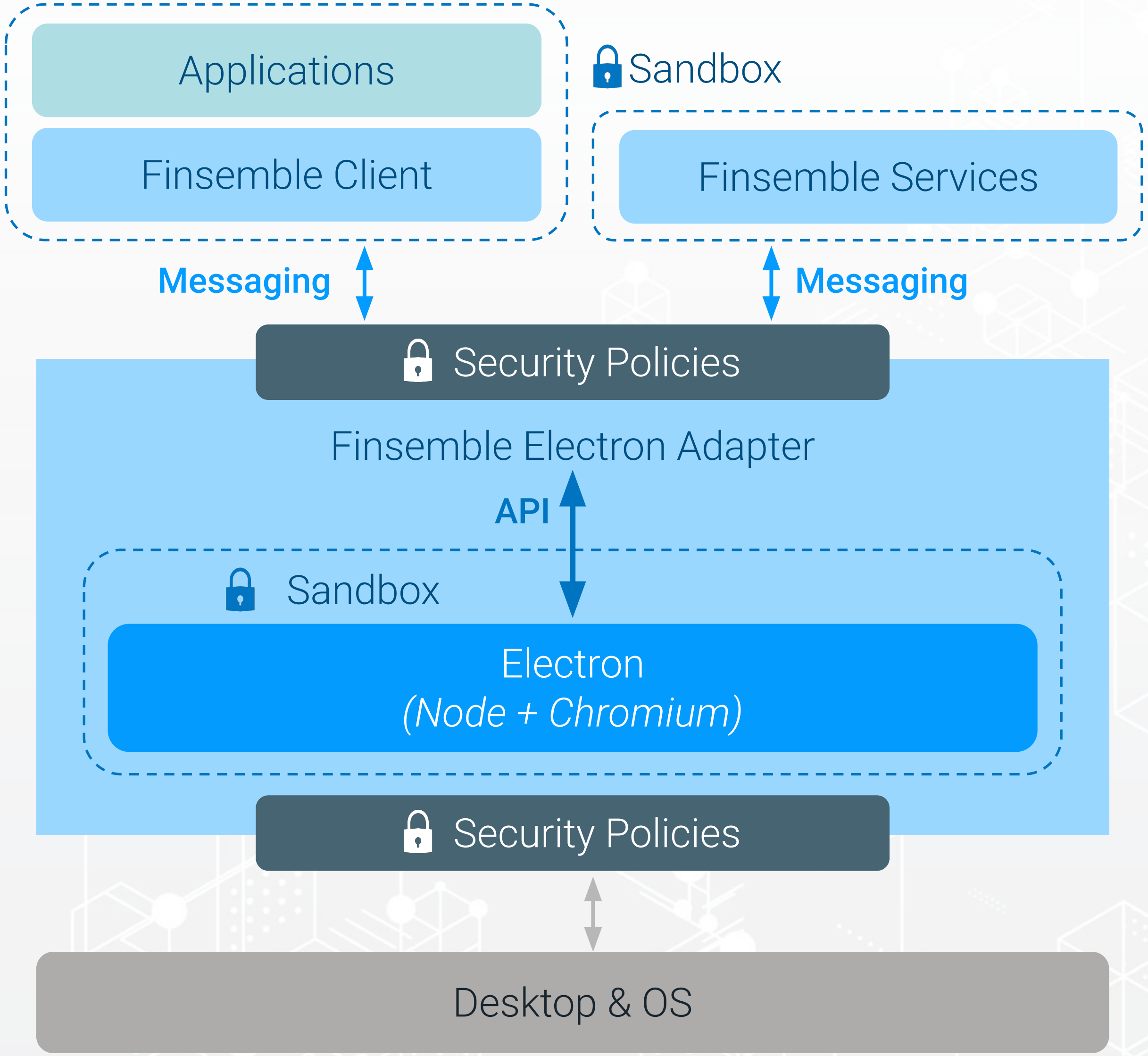
FINSEMBLE

*A fully featured Smart Desktop for finance*

www.finsemble.com

IQ

# Enterprise-secured container



- Leveraged use of Chromium, Electron & SEA
- Layers of protection, least-privilege by default
- Only vendor to provide 100% full source to clients
- Third-party security assessment by Bishop Fox

BISHOPFOX

**COMPREHENSIVE REPORT**

CHARTIQ, INC.

FINSEMBLE HYBRID APPLICATION ASSESSMENT 2019

JULY 8, 2019

BISHOP FOX

**Finding Counts**
2 Informational

**2** Total findings

**Scope**
Finsemble application

**Dates**
05/17/2019
*Kickoff*

05/17/2019 – 05/23/2019
*Active testing*

07/08/2019
*Report delivery*

# Takeaways

- Electron isn't designed to be secure out-of-the-box
- Building a 'Smart Desktop' leveraging web deployment create new risks to manage
- Electron project cares deeply about the security of your applications
  - 17-point security checklist for securing untrusted content
  - Implementing the checklist eliminates many of the benefits of using the container
- Policy-based security and Desktop Services provide the answer to practical development
- Secure Electron Adapter provides an ideal foundation to build on

# Questions?

**Thank you for attending.**

**Contact us at**
**info@chartiq.com**