Spyglass:
Secure Cloud System Administration

Patrick T. Cable II, Nabil Schear

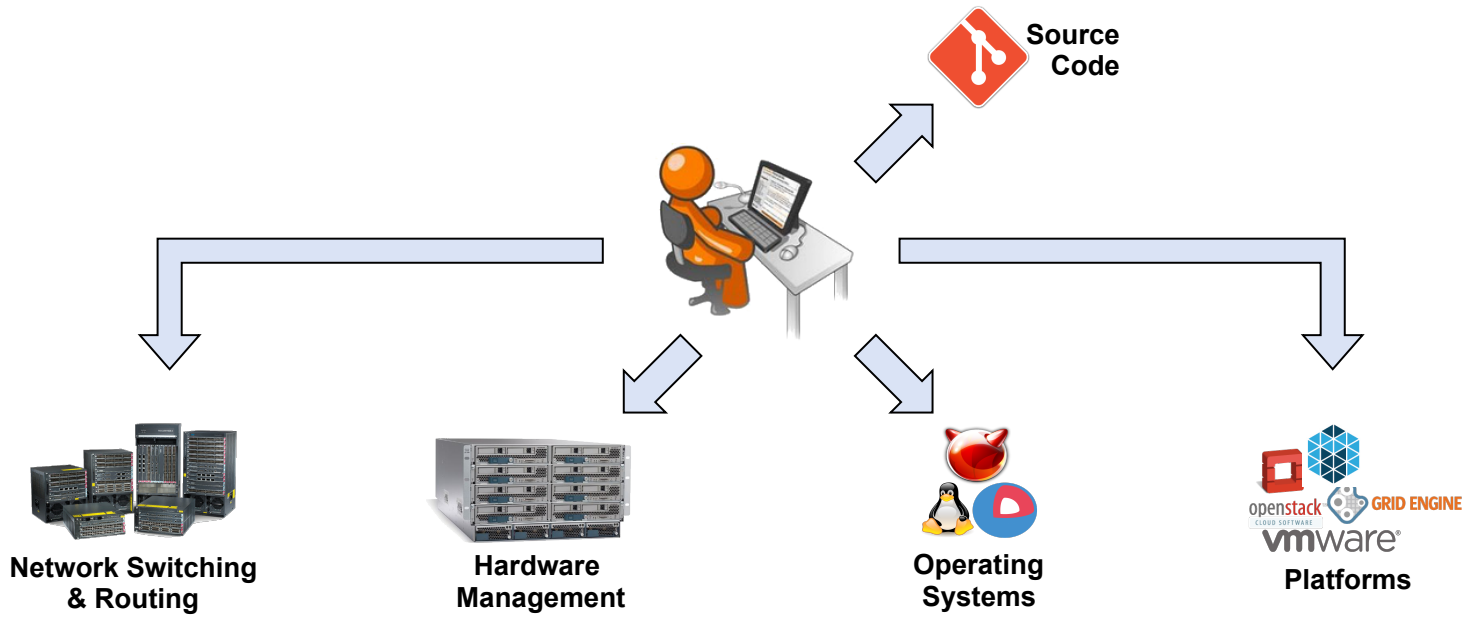
29th USENIX Large Installation System Administration Conference

11 November 2015

LINCOLN LABORATORY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Distribution Statement A: Approved for Public Release, Distribution is Unlimited. This work is sponsored by the Assistant Secretary of Defense for Research & Engineering under Air Force Contract #FA8721-05-C-0002. Opinions, interpretations, conclusions and recommendations are those of the author and are not necessarily endorsed by the United States Government.

A System Administrator's Life

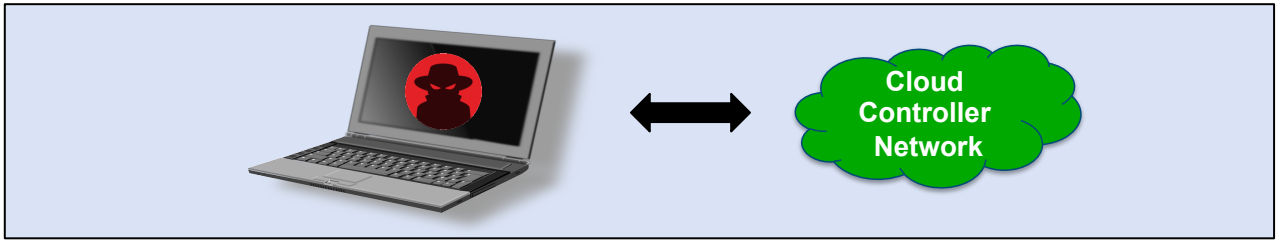


System administrators have unrestricted access to security-sensitive infrastructure

The Problem

Phishing

The Insider

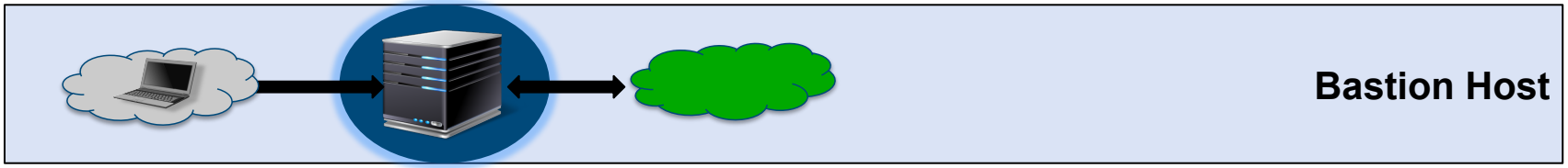


Problems for all…

… big, and small.

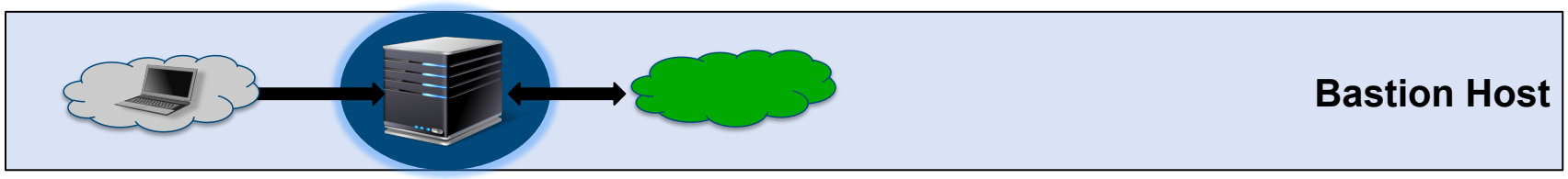
Let's Protect a Network

A look at how well different network devices audit and protect



Limit impact of malicious clients through secure auditable bastion host

Building a Better Bastion Host

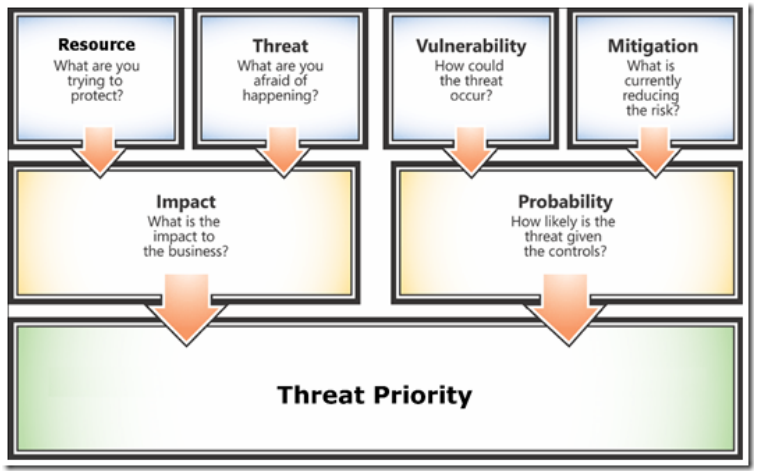


The Problem with Bastion Hosts

- Easy to implement insecurely
- Unprotected auditing
- Single point of failure
- Good for side-channel analysis

Spyglass

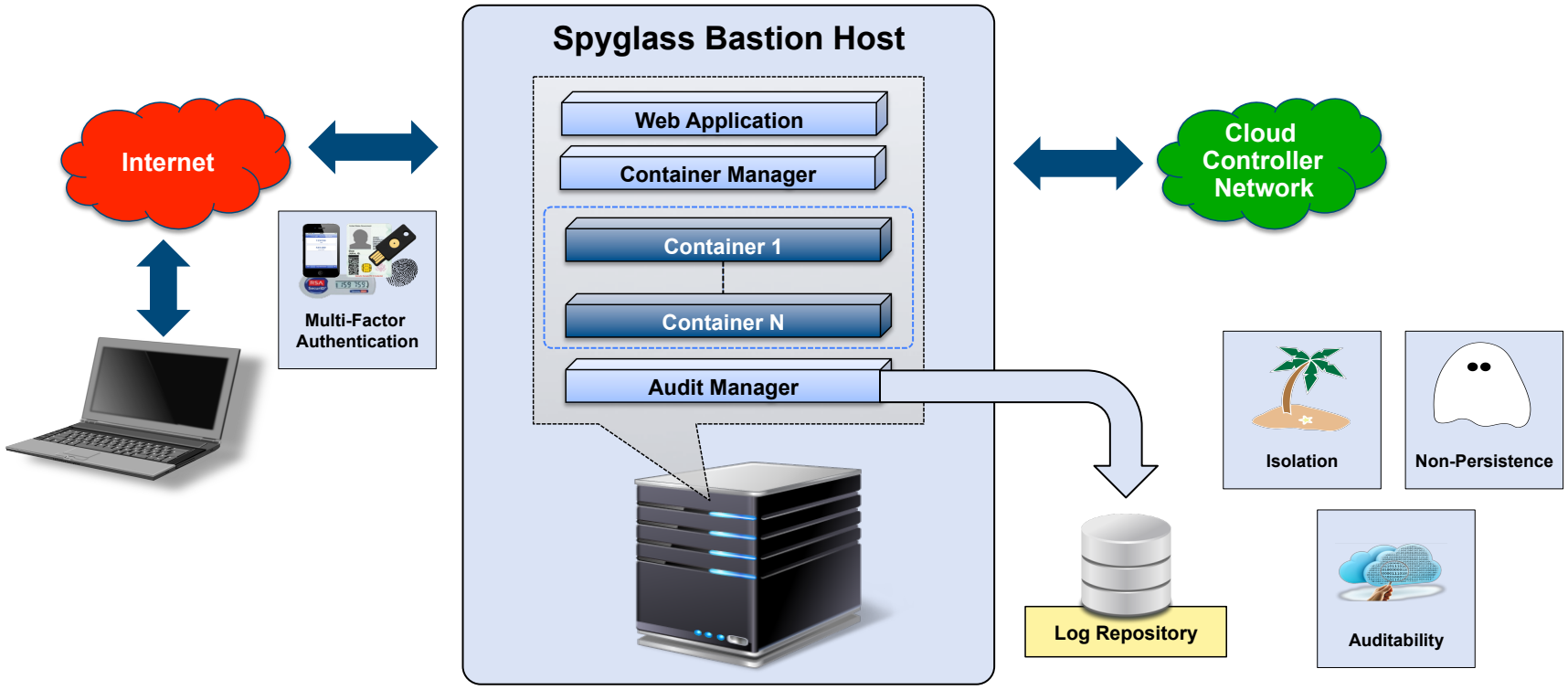
Threat Model



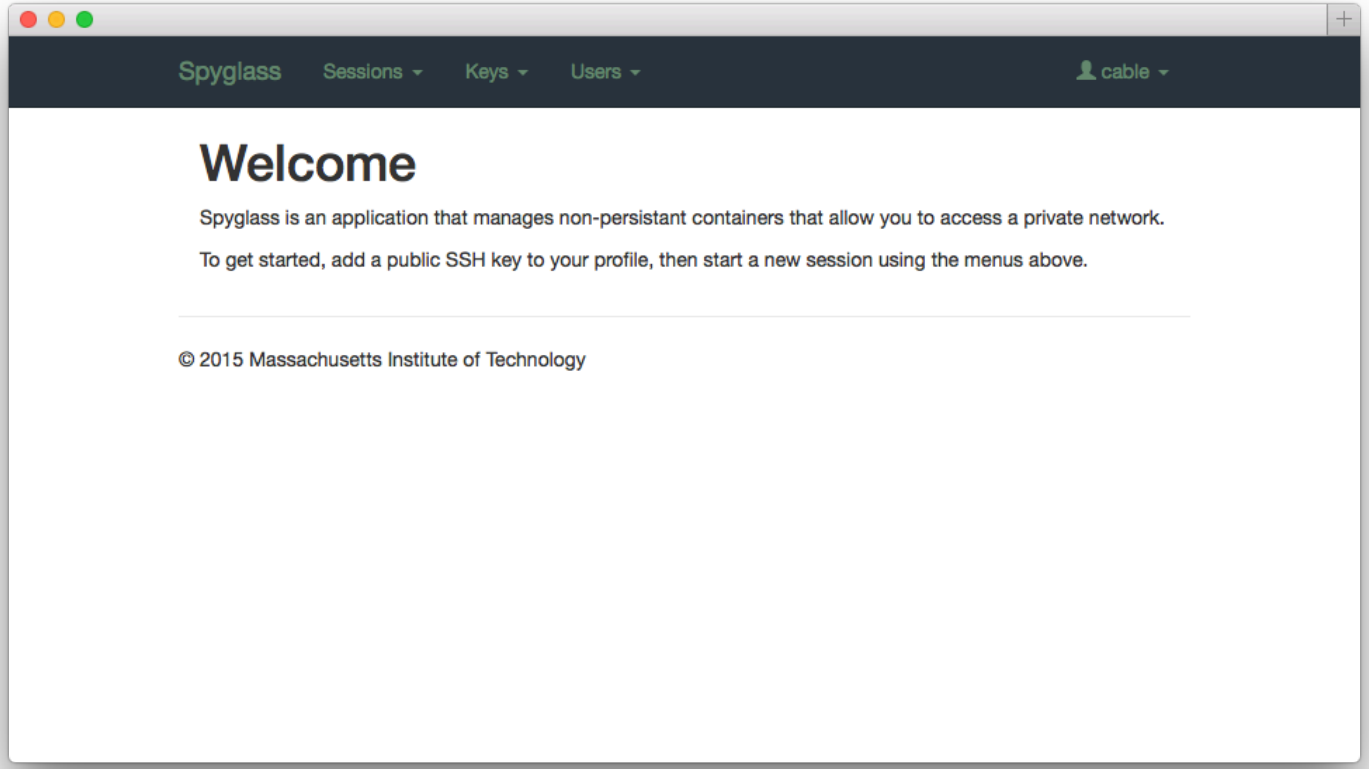
System assumptions:

- **Remote attacker trying to persistently access private network**
- **May have compromised a valid user's source system or credentials**
- **Attacker can compromise applications inside of containers that face the remote network and cannot break container isolation**
- **Attacker cannot compromise control process**
- **Proper configuration of SSH and container manager**
- **Valid users must use present multiple factors to authenticate**

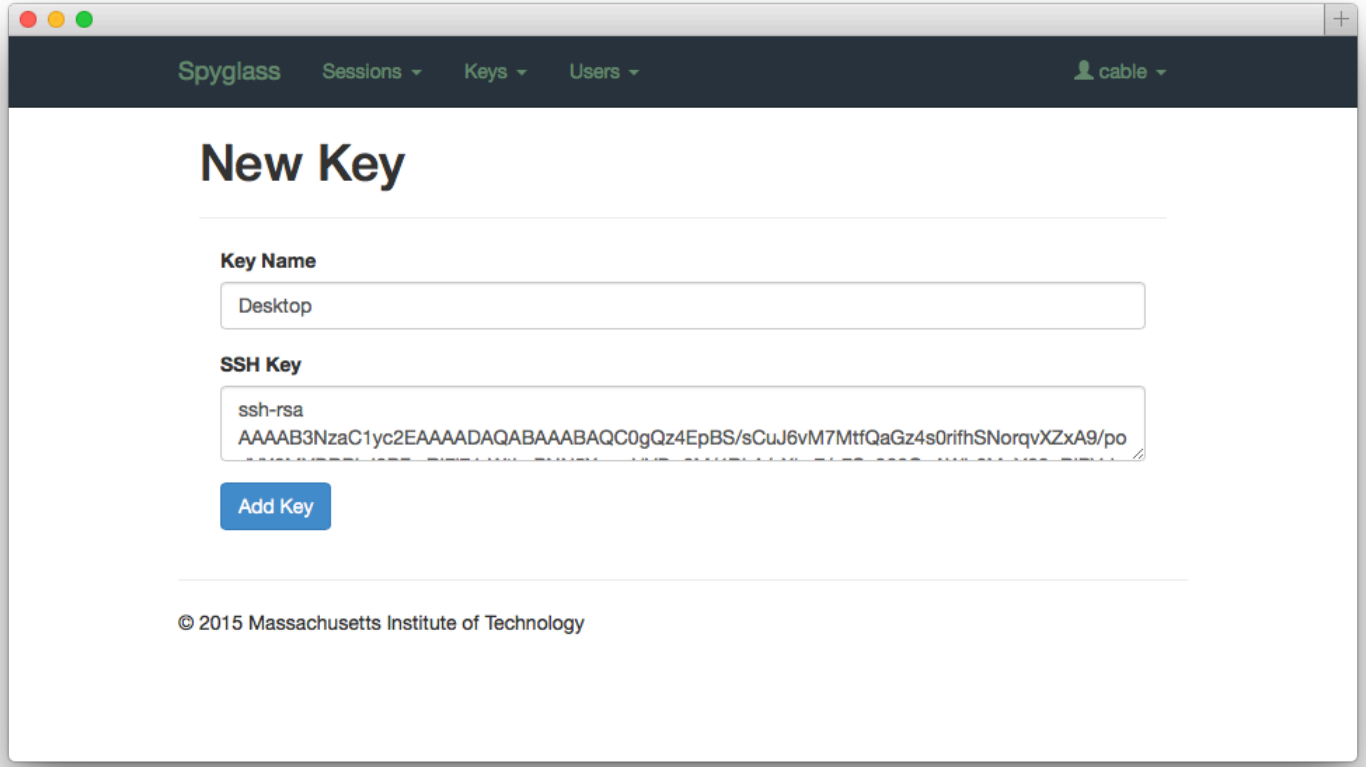
Spyglass Architecture



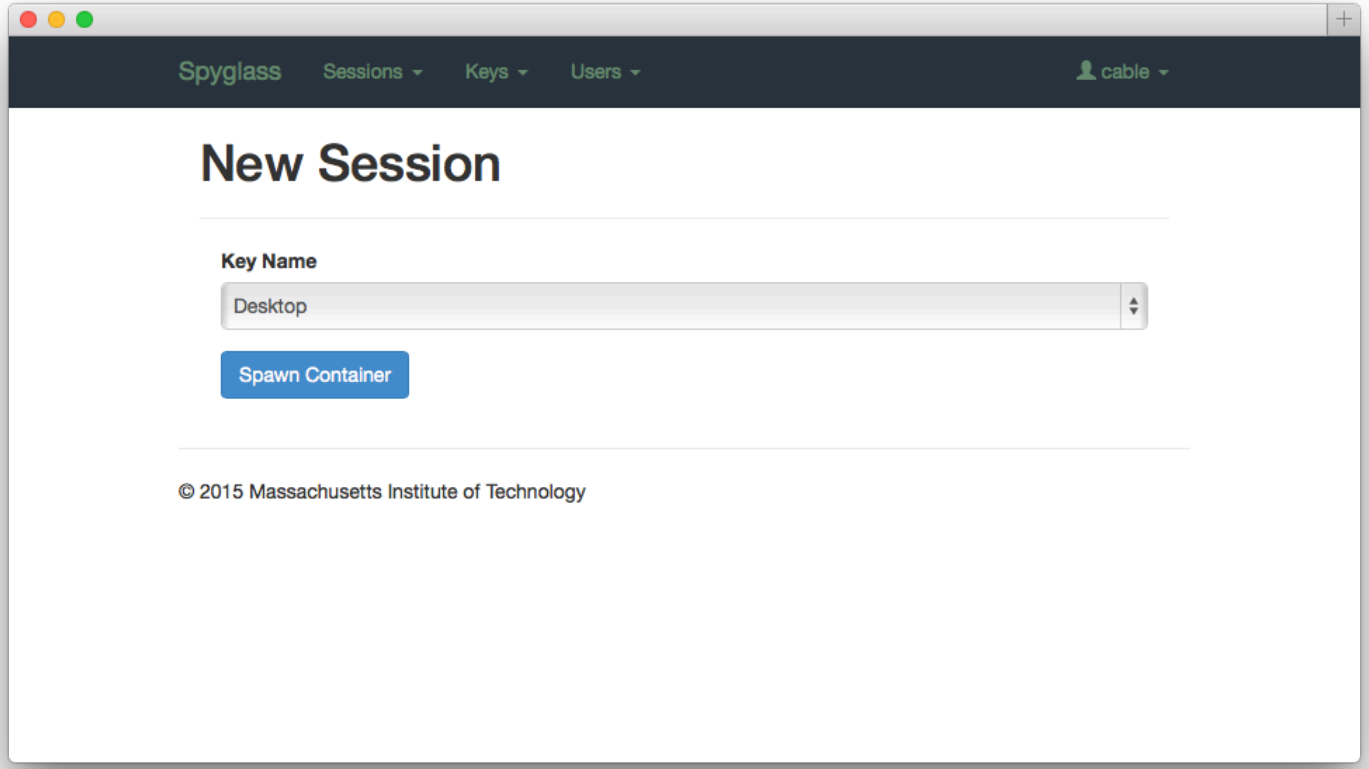
Spyglass: Login



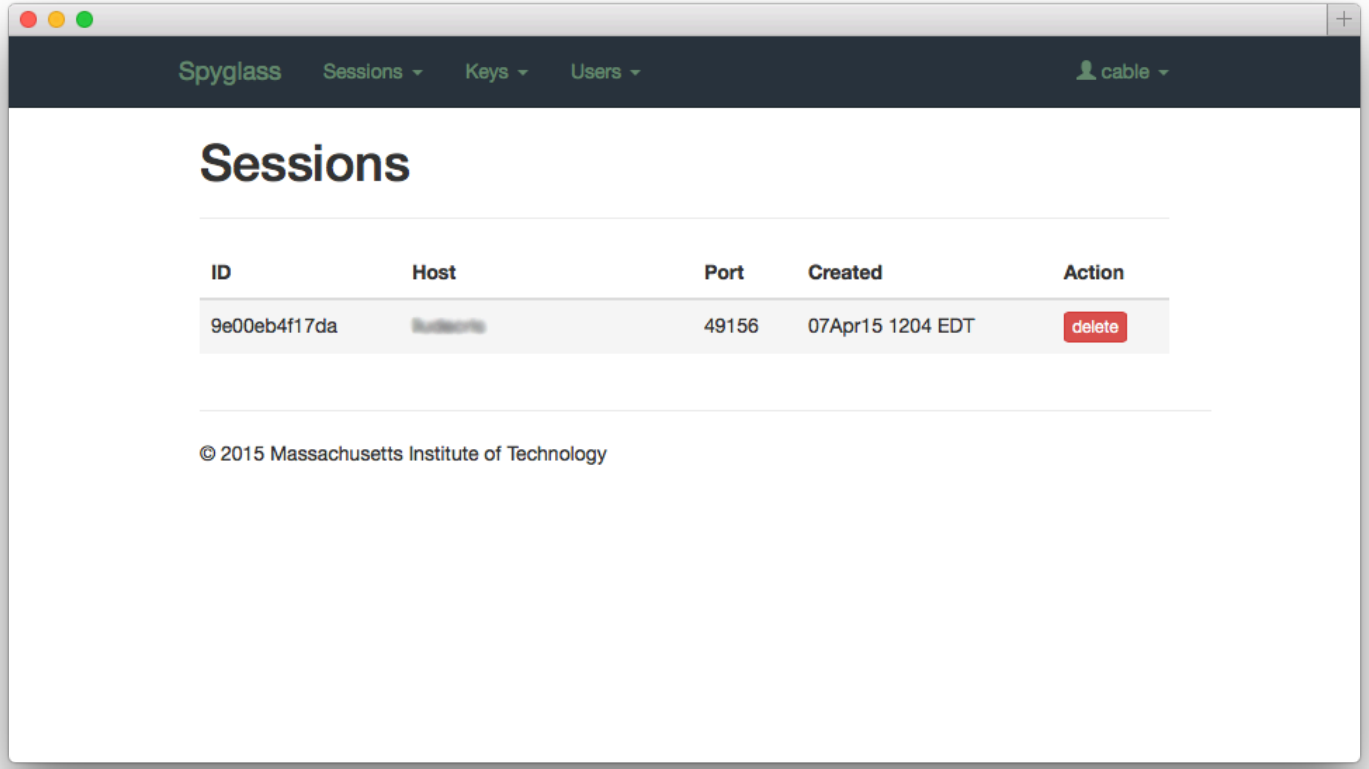
Spyglass: Add a Key



Spyglass: New Session

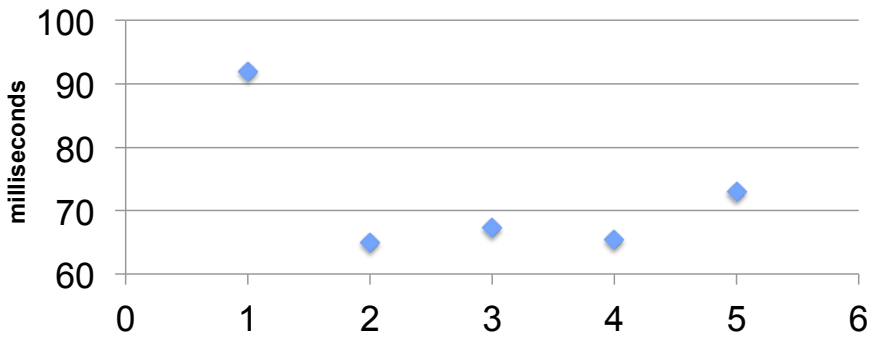


Spyglass: Session Details



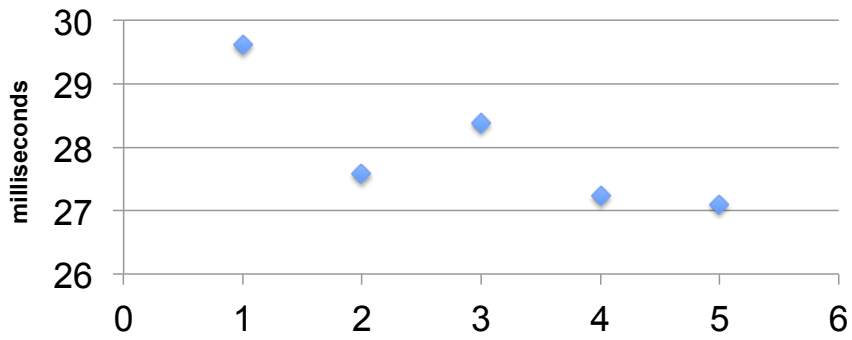
Container Instantiation Speed

Create Time



Start	**Setup** (ms)	**Container** (ms)	**Return** (ms)	**Total** (ms)
1	7.52	83.39	1.05	91.96
2	3.60	60.60	0.77	64.97
3	3.93	62.60	0.72	67.25
4	4.13	58.89	2.40	65.42
5	3.87	68.25	0.82	72.94

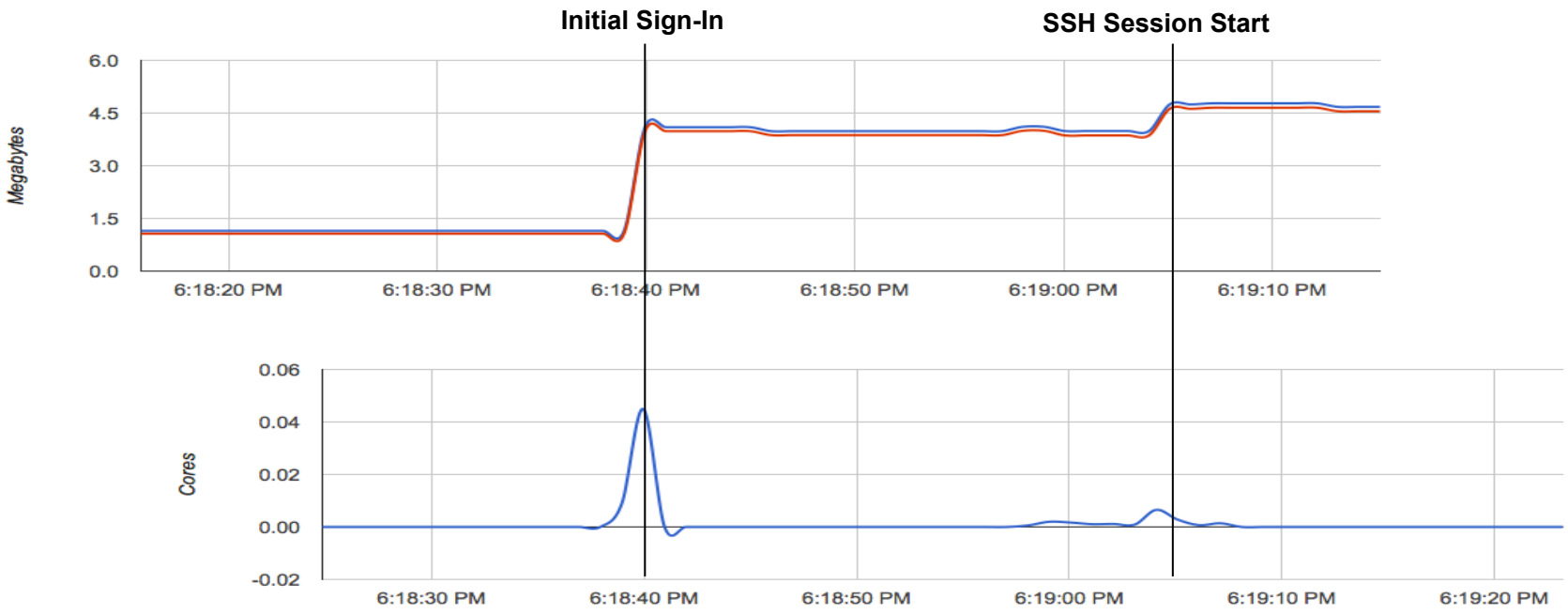
Delete Time



Stop	**Delete** (ms)	**Total** (ms)
1	25.30	29.63
2	23.54	27.59
3	24.29	28.38
4	23.04	27.24
5	22.89	27.10

Containers are quickly available for end-admin use

Host Overhead



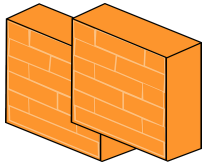
Containers are not a memory or CPU burden for the host

Attacks

Host Denial of Service

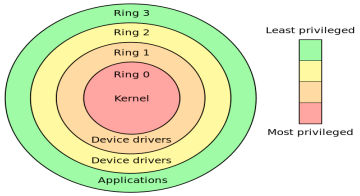
- Was able to fill file system on container host
- Workarounds noted, though may have impact on performance
- User namespaces will make this more difficult

Network Protection



- Proper configuration options with Docker disables container/ container comms
- Further tweaking with IPTables allows for finer grained controls

Escalation & Escape

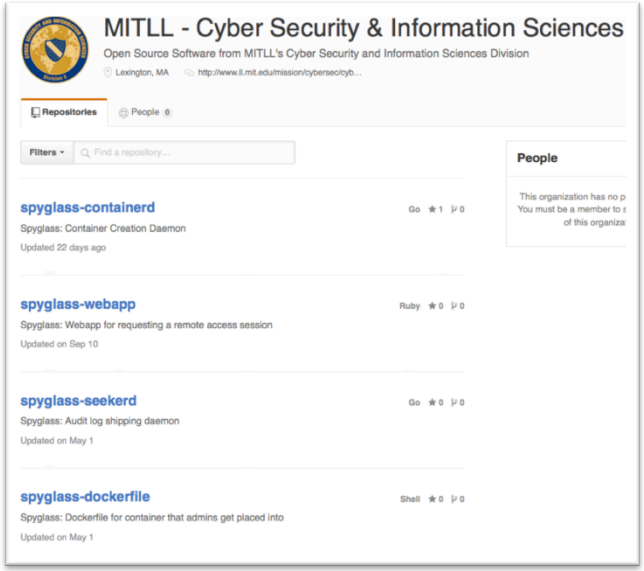


- User must escalate to root inside container
- No SELinux in demo implementation, would add another layer of complexity
- Matters to a varying degree depending on public and private networks

User Namespaces Coming Soon to Docker

Conclusion and Future Work

- **Conclusion**
  - Unauthorized access to control networks allow an attacker to wreak havoc on your organization
  - Spyglass provides an architecture to monitor your admins and protect your sensitive control networks
- **Future Work**
  - Provide container host key ID to web application
  - Make auditing collector far more resilient
  - Ignore sensitive details in audit log
  - SELinux support
  - Enterprise authentication tie-in
  - VNC session support



Fork & Improve Spyglass!
github.com/mitll-cyber

Relax system admins… you're less of a liability now!