Sandia National Laboratories

Exceptional service in the national interest

THE NATIONAL NEED FOR SOFTWARE UNDERSTANDING

Douglas Ghormley

Senior Scientist
Sandia National Laboratories

March 20, 2024

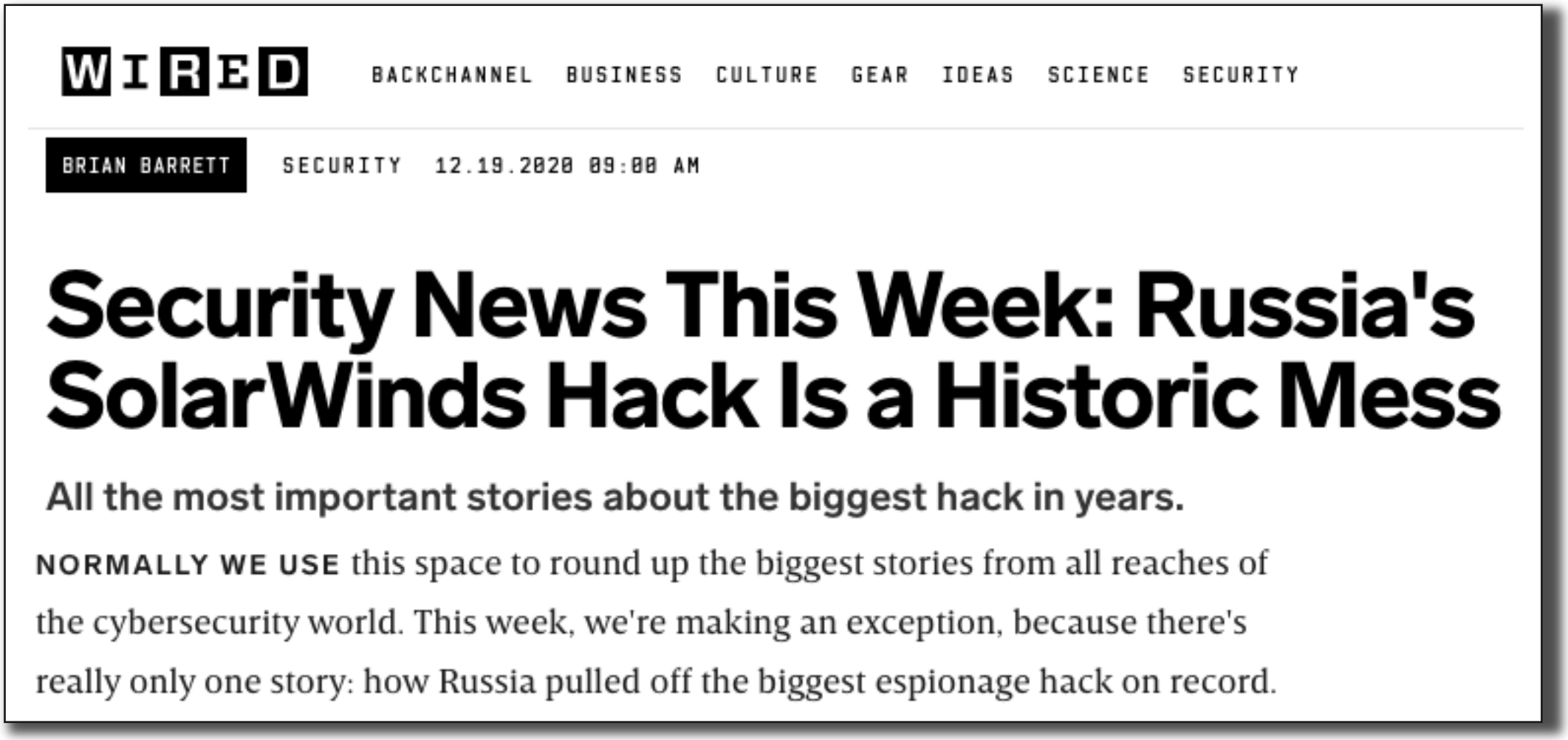
For more information on this topic, email suns@sandia.gov.

U.S. DEPARTMENT OF ENERGY | NNSA National Nuclear Security Administration

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia LLC, a wholly owned subsidiary of Honeywell International Inc. for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

SAND2024-03067PE

ACCORDING TO THE NEWS



WIRED BACKCHANNEL BUSINESS CULTURE GEAR IDEAS SCIENCE SECURITY

BRIAN BARRETT SECURITY 12.19.2020 09:00 AM

Security News This Week: Russia's SolarWinds Hack Is a Historic Mess

All the most important stories about the biggest hack in years.

NORMALLY WE USE this space to round up the biggest stories from all reaches of the cybersecurity world. This week, we're making an exception, because there's really only one story: how Russia pulled off the biggest espionage hack on record.

ACCORDING TO THE NEWS

ZDNet

Microsoft confirms it was also breached in recent SolarWinds supply chain hack

Microsoft denies that hackers pivoted to production systems and abused its software to attack customers.

By Catalin Cimpanu for Zero Day | December 17, 2020 -- 23:46 GMT (15:46 PST) | Topic: Security

The vast majority of these victims are US government agencies, such as:

- The US Treasury Department
- The US Department of Commerce's National Telecommunications and Information Administration (NTIA)
- The Department of Health's National Institutes of Health (NIH)
- The Cybersecurity and Infrastructure Agency (CISA)
- The Department of Homeland Security (DHS)
- The US Department of State
- The National Nuclear Security Administration (NNSA) (*also disclosed today*)
- The US Department of Energy (DOE) (*also disclosed today*)
- Three US states (*also disclosed today*)
- City of Austin (*also disclosed today*)

ACCORDING TO THE NEWS

ZDNet

Microsoft confirm… also breached in recent SolarWind…

Microsoft denies that hackers pivoted to …

By Catalin Cimpanu for Zero Day | De…
(15:46 PST) | Topic: Security

The vast majority of these victims…

- The US Treasury Department
- The US Department of Commerce's National Telecommunications… Administration (NTIA)
- The Department of Health's National Institutes of Health (NIH)
- The Cybersecurity and Infrastructure Agency (CISA)
- The Department of Homeland Security (DHS)
- The US Department of State
- The National Nuclear Security Administration (NNSA) (*also disclosed today*)
- The US Department of Energy (DOE) (*also disclosed today*)
- Three US states (*also disclosed today*)
- City of Austin (*also disclosed today*)

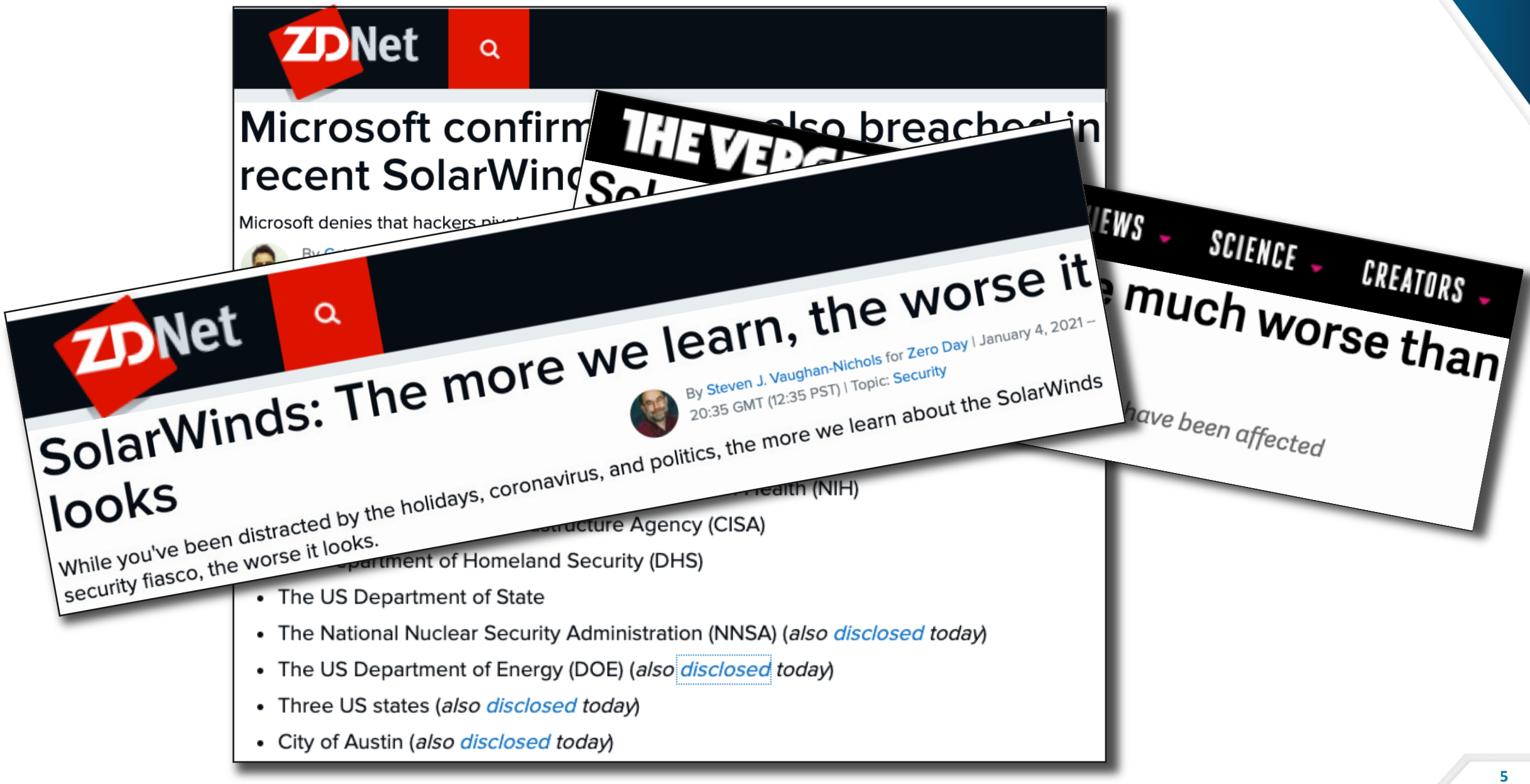
THE VERGE TECH REVIEWS SCIENCE CREATORS

SolarWinds hack may be much worse than originally feared

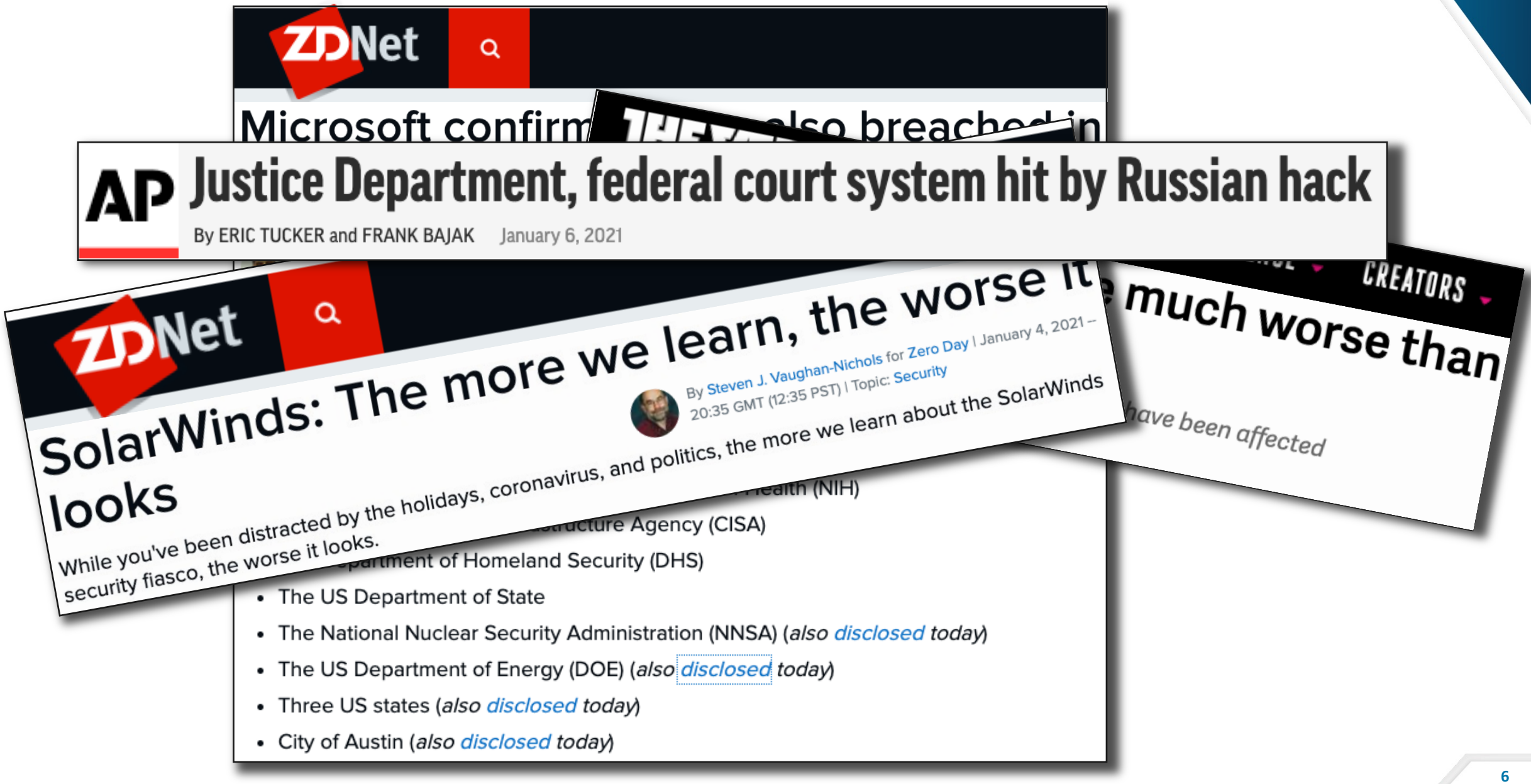
Some 250 government agencies and businesses may have been affected

By Kim Lyons | Jan 2, 2021, 4:50pm EST

ACCORDING TO THE NEWS



ACCORDING TO THE NEWS



ACCORDING TO THE NEWS

If it were your job
to detect such things in software
before it is put into use,
what would you do?

WAYS OF DEALING WITH SOFTWARE RISK TODAY

Ask the Developer

Run Some Tests

Check the Signature

Investigate

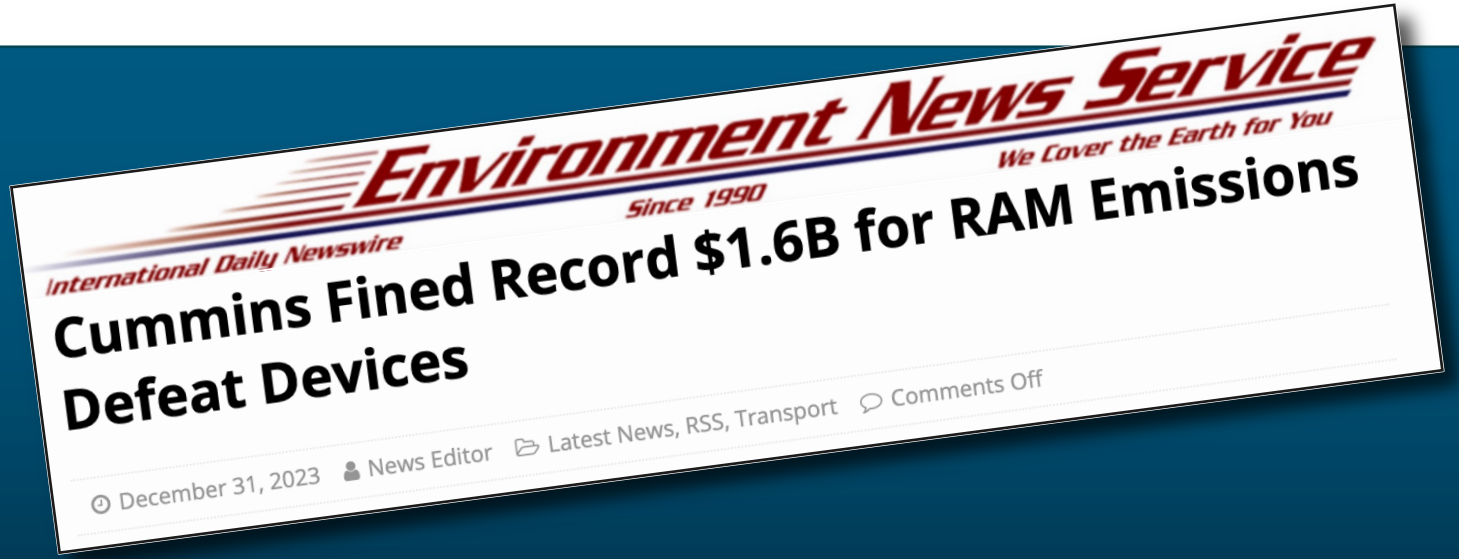
WAYS OF DEALING WITH SOFTWARE RISK TODAY

Ask the Developer

Run Some Tests

Check the Signature

Investigate



WAYS OF DEALING WITH SOFTWARE RISK TODAY

Ask the Developer

Run Some Tests

Check the Signature

Investigate

WAYS OF DEALING WITH SOFTWARE RISK TODAY

Ask t[illegible]eloper

Run Some Tests

Check the Signature

Investigate

Contractor admits plant[illegible]

International Daily

Cummin

Defeat D

December 31, 2023

WIRED BACKCHANNEL BUSINESS CULTURE GEAR IDEAS POLITICS SCIENCE SECURITY MERCH

KIM ZETTER SECURITY JUL 24, 2023 6:00 AM

Code Kept Secret for Years Reveals Its Flaw—a Backdoor

A secret encryption cipher baked into radio systems used by critical infrastructure workers, police, and others around the world is finally seeing sunlight. Researchers say it isn't pretty.

bombs

t new

WAYS OF DEALING WITH SOFTWARE RISK TODAY

Ask t[illegible]eloper

Run Some Tests

Check the Signature

Investigate

Evasive Malware Detects and Defeats Virtual Machine Analysis

POSTED BY **LASTLINE** ON OCT 24, 2016

Advanced malware solutions ("sandboxes") traditionally use virtual machines (VM) to analyze suspicious objects to find out if they are malicious. However, advanced malware is capable of detecting the presence of the virtual machine technology used by conventional sandboxes and leveraging this weakness to evade detection.

WAYS OF DEALING WITH SOFTWARE RISK TODAY

Ask t[illegible]eloper

Run Some Tests

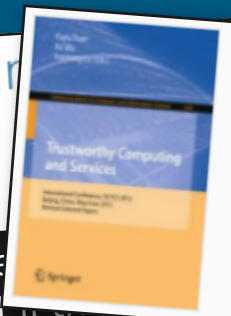
Check the Signature

Investigate

Evasive Malware Detects ar[illegible] Machine Analysis

POSTED BY **LASTLINE** ON OCT 24, 2016

Advanced malware solutions ("sandboxe[illegible] to analyze suspicious objects to find out [illegible] malware is capable of detecting the presence of the virtual [illegible] used by conventional sandboxes and leveraging this weakness to evade detect[illegible]



International Conference on Trustworthy Computing and Services

ISCTCS 2012: Trustworthy Computing and Services pp 34-44 | Cite as

Software Testing is Necessary But Not Sufficient for Software Trustworthiness

WAYS OF DEALING WITH SOFTWARE RISK TODAY

Ask t[illegible]eloper

Run S[illegible]ests

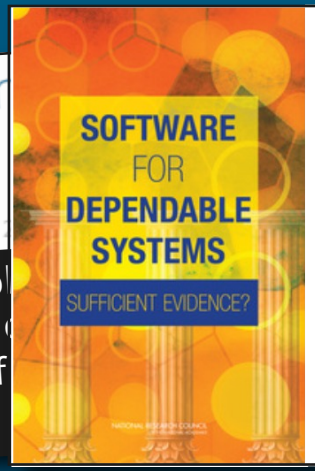
Check the Signature

Investigate

Evasive Malwar
Analysis

POSTED BY LASTLINE ON OCT

Advanced malware so
to analyze suspicious
malware is capable of
used by conventional



Software for Dependable Systems (2007)
National Research Council

...it is important to realize that testing alone is rarely sufficient to establish high levels of dependability.

as

t Not Sufficient for

WAYS OF DEALING WITH SOFTWARE RISK TODAY

Ask t[illegible]eloper

Run S[illegible]ests

Check the Signature

Investigate

DARKREADING

Dave Roche

August 1, 2023

Lessons Not Learned From Software Supply Chain Attacks

In December, an unauthorized user accessed GitHub's systems and stole three encrypted code-signing certificates: one Apple-issued Developer ID certificate and two DigiCert-issued code-signing certificates for its desktop and Atom applications.

Another security breach at Micro-Star International (MSI) resulted in a software supply chain attack, where hackers had access to private signing keys for MSI's firmware and Intel's UEFI.

WAYS OF DEALING WITH SOFTWARE RISK TODAY

Ask t[illegible]eloper

Run S[illegible]ests

Check the Signature

Investigate

DARKREADING

Hackers are selling legitimate code-signing certificates to evade malware detection

Code-signed apps are harder to detect by network security appliances, making it easier to sneak malware onto a vulnerable system. The downside? Certificates aren't cheap — and hackers usually are.

By Zack Whittaker for Zero Day | February 22, 2018 -- 13:00 GMT (05:00 PST) | Topic: Security

Dave Roche

August 1, 2023

[illegible]in Attacks

[illegible] code-signing [illegible]g certificates for

[illegible] resulted in a software supply chain attack, where [illegible] private signing keys for MSI's firmware and Intel's UEFI.

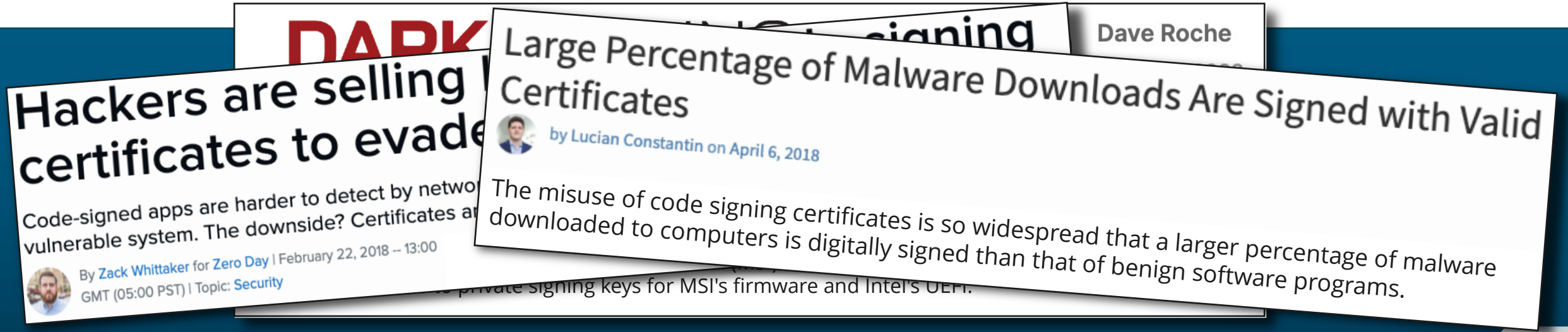
WAYS OF DEALING WITH SOFTWARE RISK TODAY

Ask t[illegible]eloper

Run S[illegible]ests

Check [illegible]gnature

Investigate



WAYS OF DEALING WITH SOFTWARE RISK TODAY

Ask t[illegible]eloper

Run S[illegible]ests

Check [illegible]gnature

Investigate

Tens of millions of biz Dell PCs smacked by privilege-escalation bug in bundled troubleshooting tool

If you don't have auto-update switched on, time to patch

SHARE

Laurie Clarke Tue 11 Feb 2020 // 15:01 UTC

WAYS OF DEALING WITH SOFTWARE RISK TODAY

Ask t[illegible]eloper

Run S[illegible]ests

Check [illegible]gnature

Investigate

Tens of millions of biz Dell P
in bundled troubleshooting

If you don't have auto-update switched

Laurie Clarke Tue 11 Feb 2020 // 15:01 UTC

[illegible]escalation bug

Chinese Hackers Exploit Cisco, Citrix Flaws in Massive Espionage Campaign

Between Jan. 20 and March 11, researchers observed APT41 exploiting vulnerabilities in Citrix NetScaler/ADC, Cisco routers and Zoho ManageEngine Desktop Central as part of the widespread espionage campaign.

Author:
Lindsey O'Donnell
March 25, 2020 / 11:57 am

WAYS OF DEALING WITH SOFTWARE RISK TODAY

Ask t[illegible]eloper

Run S[illegible]ests

Check [illegible]gnature

In[illegible]te

THE SOFTWARE SUPPLY CHAIN IS GLOBAL

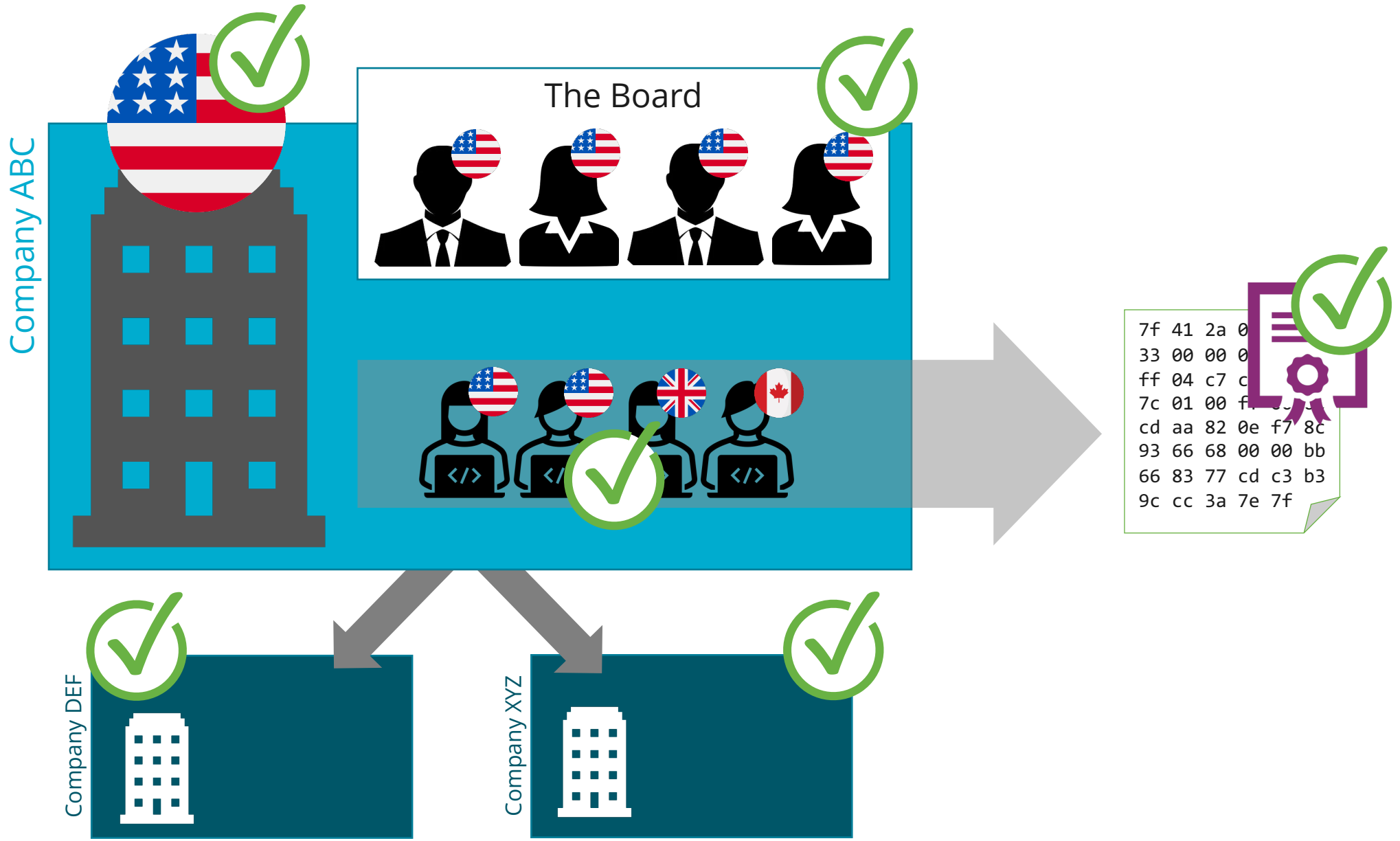
Source: An advertising video from the Czech Republic encouraging the outsourcing of code development

THE SOFTWARE SUPPLY CHAIN IS GLOBAL

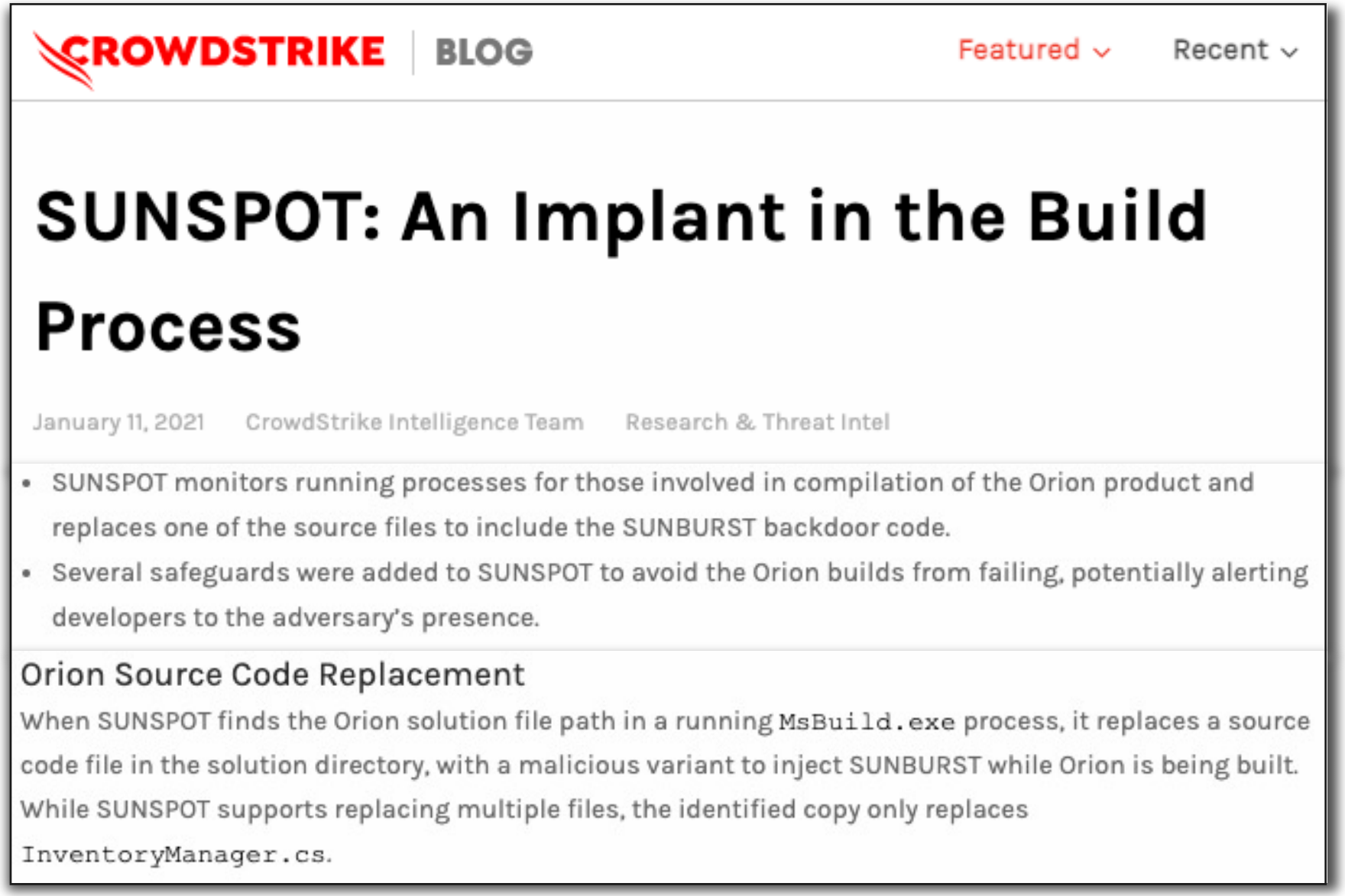
In a global economy it is infeasible
to only use software developed
in “trustworthy” countries.

Source: An advertising video from the Czech Republic encouraging the outsourcing of code development

SUPPLY CHAIN ASSURANCE – A MAJOR GAP



SUPPLY CHAIN ASSURANCE – A MAJOR GAP



CROWDSTRIKE | BLOG

Featured Recent

SUNSPOT: An Implant in the Build Process

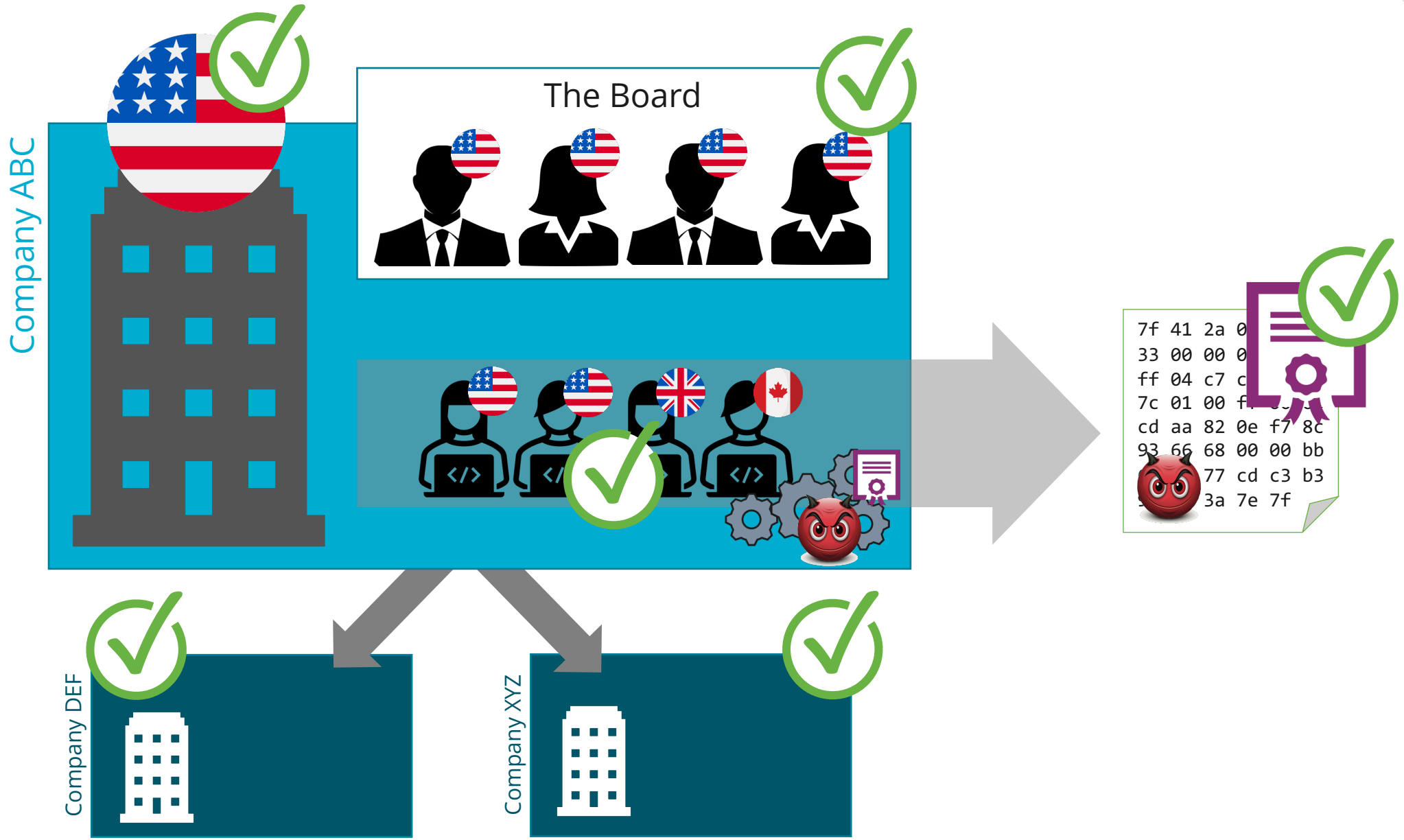
January 11, 2021 CrowdStrike Intelligence Team Research & Threat Intel

- SUNSPOT monitors running processes for those involved in compilation of the Orion product and replaces one of the source files to include the SUNBURST backdoor code.
- Several safeguards were added to SUNSPOT to avoid the Orion builds from failing, potentially alerting developers to the adversary's presence.

Orion Source Code Replacement

When SUNSPOT finds the Orion solution file path in a running `MsBuild.exe` process, it replaces a source code file in the solution directory, with a malicious variant to inject SUNBURST while Orion is being built. While SUNSPOT supports replacing multiple files, the identified copy only replaces `InventoryManager.cs`.

SUPPLY CHAIN ASSURANCE – A MAJOR GAP



SUPPLY CHAIN ASSURANCE – A MAJOR GAP

NEWS

Russia's SolarWinds hack has no easy fix, cybersecurity company says

Efforts to assess the impact of a more than seven-month-old cyberespionage campaign blamed on Russia – and boot the intruders – remain in their early stages.

Jan. 19, 2021, 9:59 AM MST / Updated Jan. 19, 2021, 10:01 AM MST

By The Associated Press

Carmakal said he believed software companies were prime targets because hackers of this caliber will seek to use their products — as they did with SolarWinds' Orion module — as conduits for similar so-called supply-chain hacks.

https://www.nbcnews.com/tech/security/russias-solarwinds-hack-no-easy-fix-cybersecurity-company-says-rcna227

WHAT'S NEEDED MORE THAN ALL OF THESE?

Ask the Developer

Run Some Tests

Check the Signature

Investigate

Analyze the Software's Behavior!

Assess the *potential behavior* of the software itself
to answer key national security & critical infrastructure mission questions.

WHAT'S NEEDED MORE THAN ALL OF THESE?

Ask the Developer

Run Some Tests

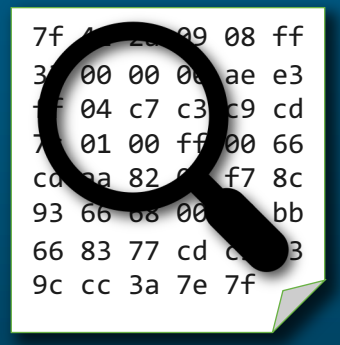
Check the Signature

Investigate

All of the questions above are proxies for the one thing that matters most:

What can the software actually do?

Confidently answering questions about existing software requires technical analysis of the potential behaviors of the software.



THE NATIONAL NEED

The USG has integrated 3rd-party software into every facet of national security (NS), critical infrastructure (CI), and government.

This software regularly exhibits undesirable behaviors that put mission at risk.

To ensure NS and CI mission success, we must pose and answer a variety of mission-specific questions about software's potential behavior.

“MISSION QUESTIONS” ABOUT SOFTWARE IN A DESTROYER

Could sensitive communications be relayed off-ship?

Will the radar software lie under certain conditions?

Is there hidden, triggered behavior in our weapon’s targeting systems?

Could false commands be issued by this software?

Can my secure comms software send unencrypted messages?

Can the software leak my ship’s location?

“MISSION QUESTIONS” ABOUT SOFTWARE ACROSS MISSIONS

Does this software have a backdoor?

What indicators does this malware have?

Is this software vulnerable to attack XYZ?

Could our sensitive data be leaked?

Does this software have hidden ransomware?

Could our sensitive data be changed without aurhotization?

THE FULL SCOPE OF THE PROBLEM

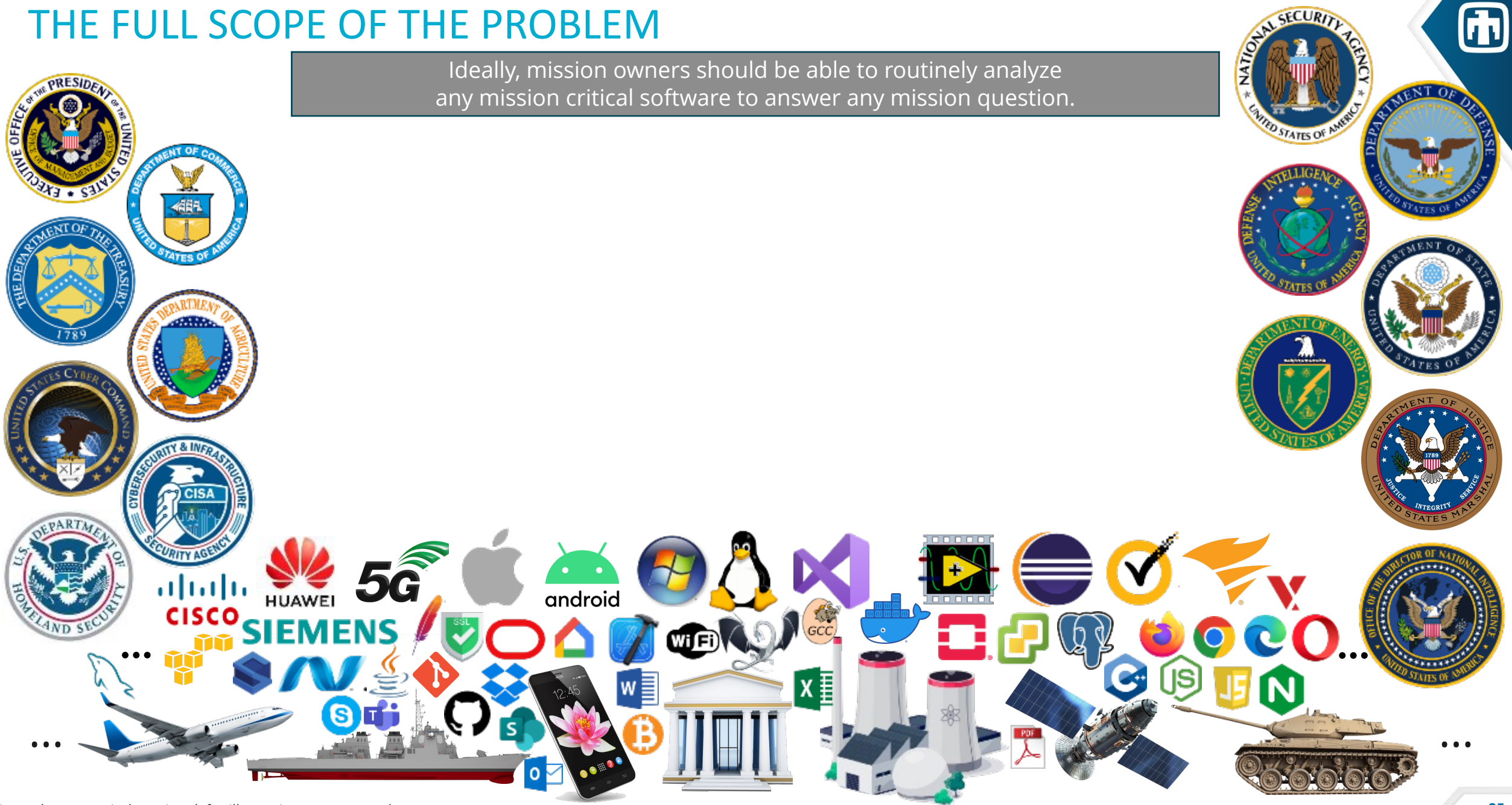
Examples are entirely notional, for illustration purposes only.

THE FULL SCOPE OF THE PROBLEM

Examples are entirely notional, for illustration purposes only.

THE FULL SCOPE OF THE PROBLEM

Ideally, mission owners should be able to routinely analyze
any mission critical software to answer any mission question.



Examples are entirely notional, for illustration purposes only.

THE FULL SCOPE OF THE PROBLEM

Ideally, mission owners should be able to routinely analyze any mission critical software to answer any mission question.

Could our sensitive data be changed without authorization?

Does this software have hidden ransomware?

Is there a kill switch hidden in my propulsion systems?

What indicators does this malware have?

Is there hidden, triggered behavior in our weapon's targeting systems?

Could sensitive communications be relayed off-ship?

Does this software have a backdoor?

Can my ship's location be tracked?

Is this software vulnerable to attack XYZ?

Could false commands be issued by this software?

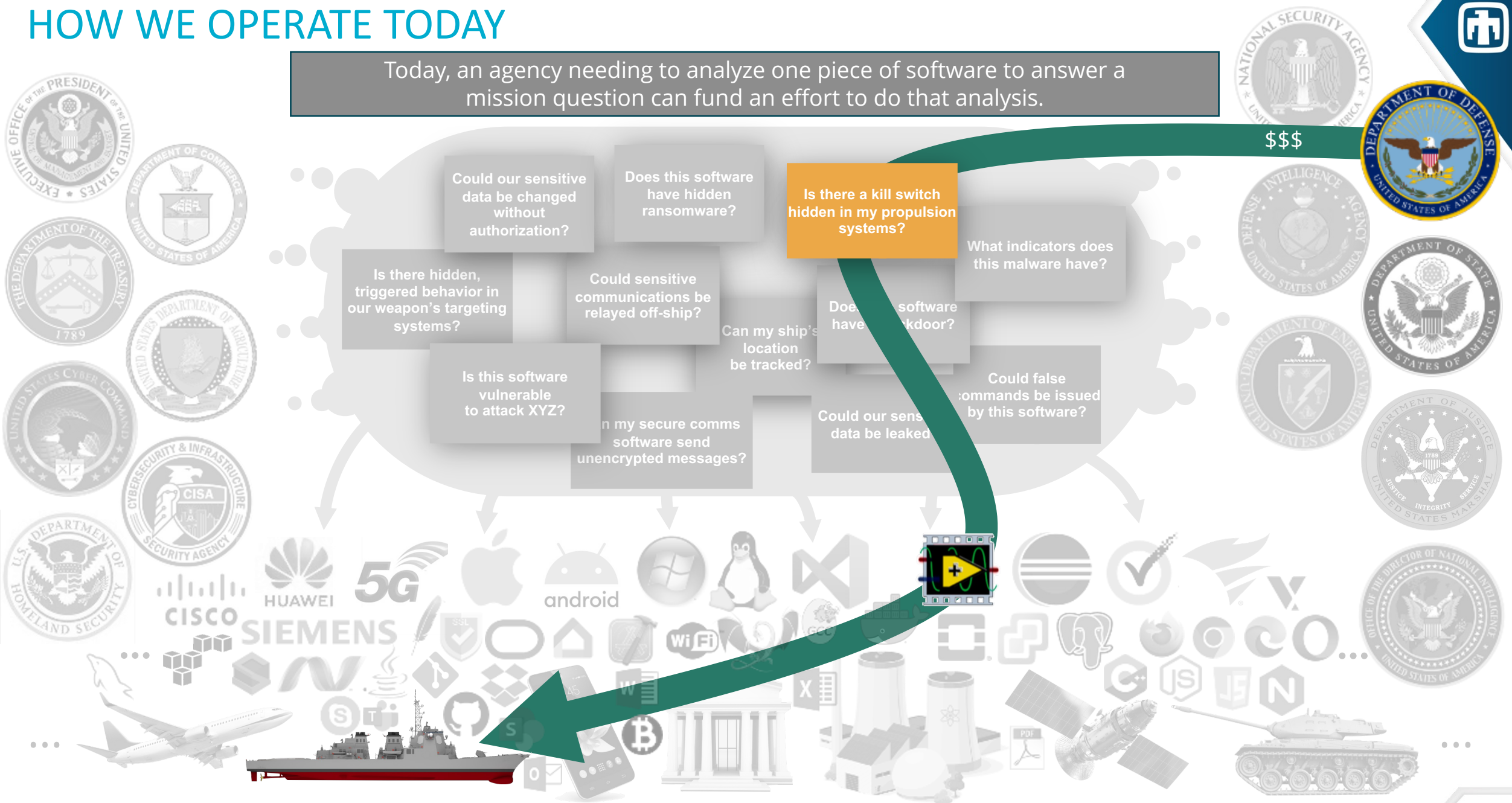
n my secure comms software send unencrypted messages?

Could our sensitive data be leaked?

Examples are entirely notional, for illustration purposes only.

HOW WE OPERATE TODAY

Today, an agency needing to analyze one piece of software to answer a mission question can fund an effort to do that analysis.



Examples are entirely notional, for illustration purposes only.

HOW WE OPERATE TODAY

Today, an agency needing to analyze one piece of software to answer a mission question can fund an effort to do that analysis.

$$$

Could our sensitive data be changed without authorization?

Does this software have hidden ransomware?

Is there a kill switch hidden in my propulsion systems?

What indicators does this malware have?

Is there hidden, triggered behavior in our weapon's targeting systems?

Could sensitive communications be relayed off-ship?

Can my ship's location be tracked?

Is this software

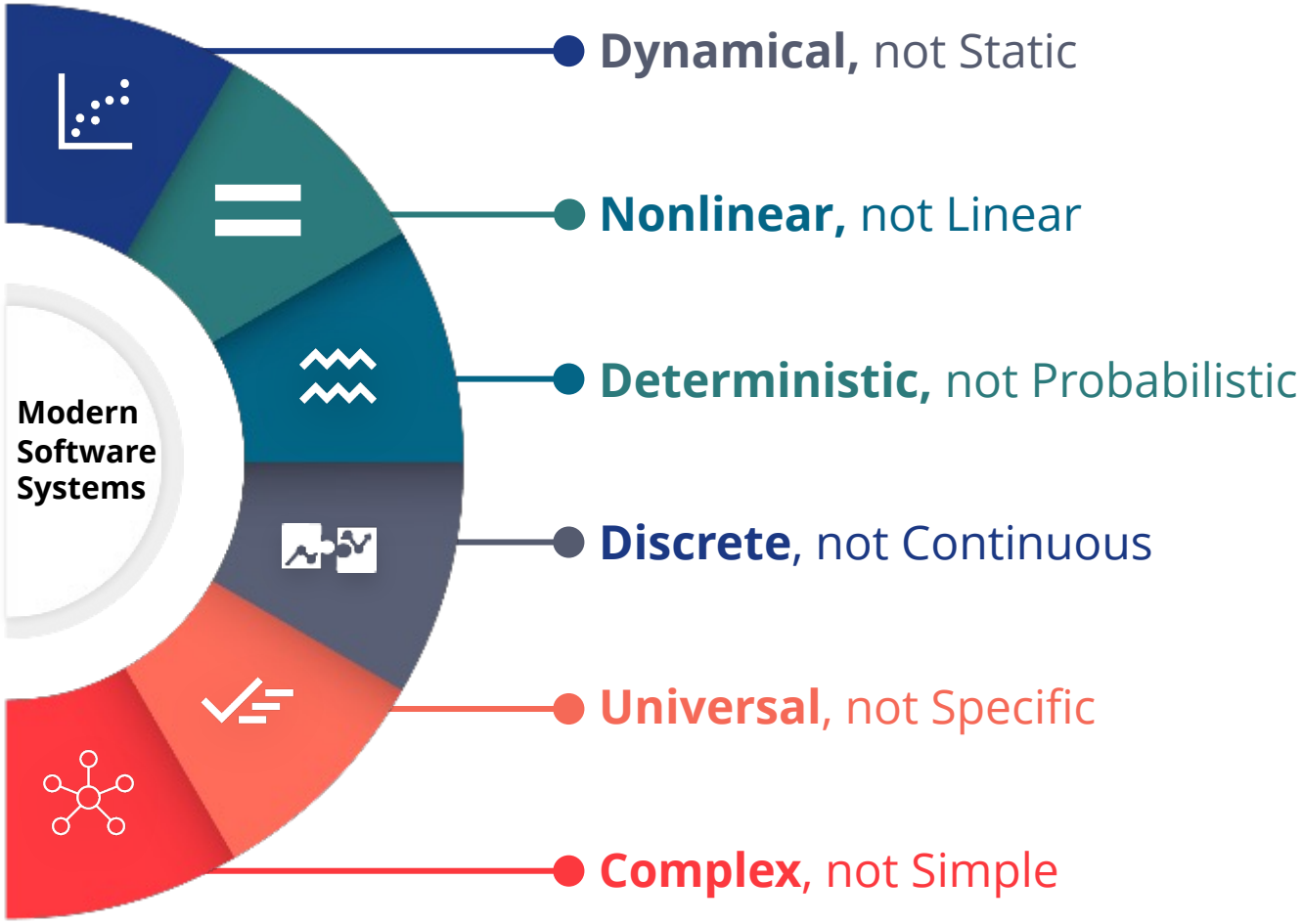
Could our sens data be leaked

Could false commands be issued by this software?

This is a major undertaking.

Examples are entirely notional, for illustration purposes only.

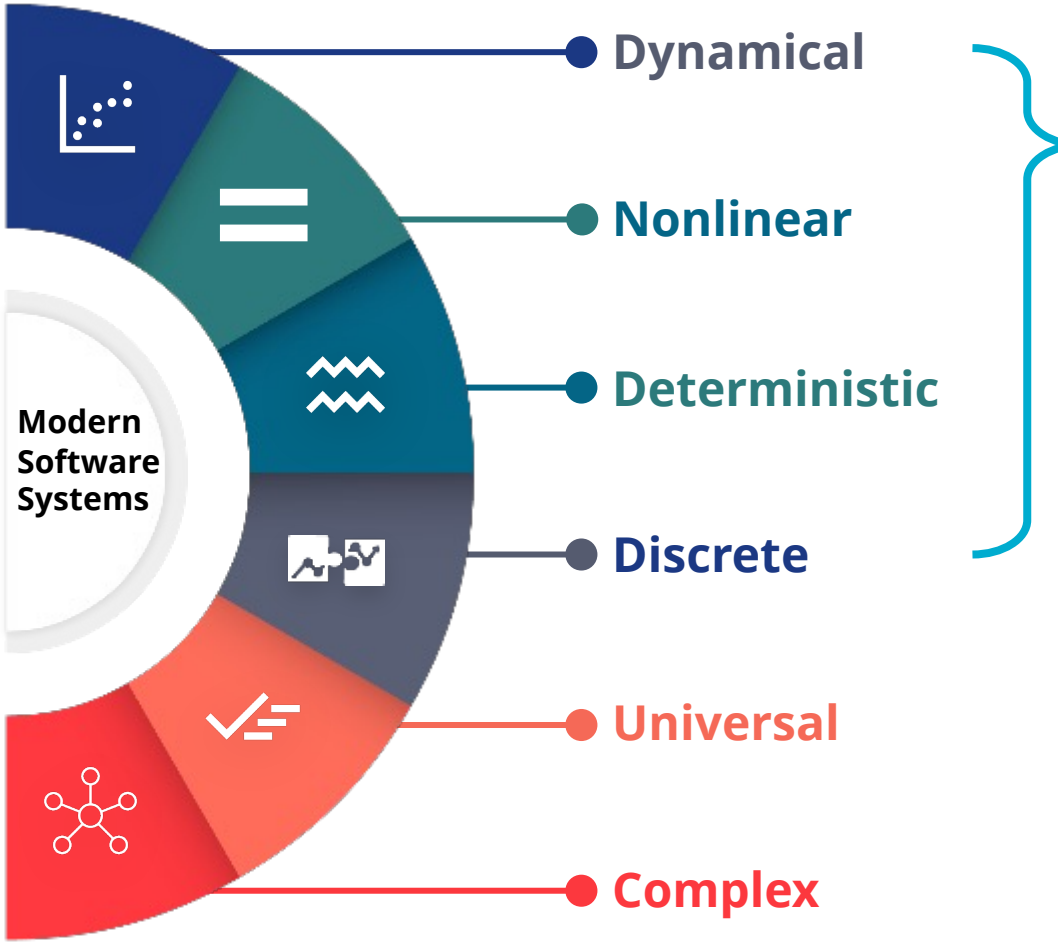
CHARACTERISTICS OF MODERN SOFTWARE SYSTEMS



This combination of characteristics is the hardest of the set of options for analysis.

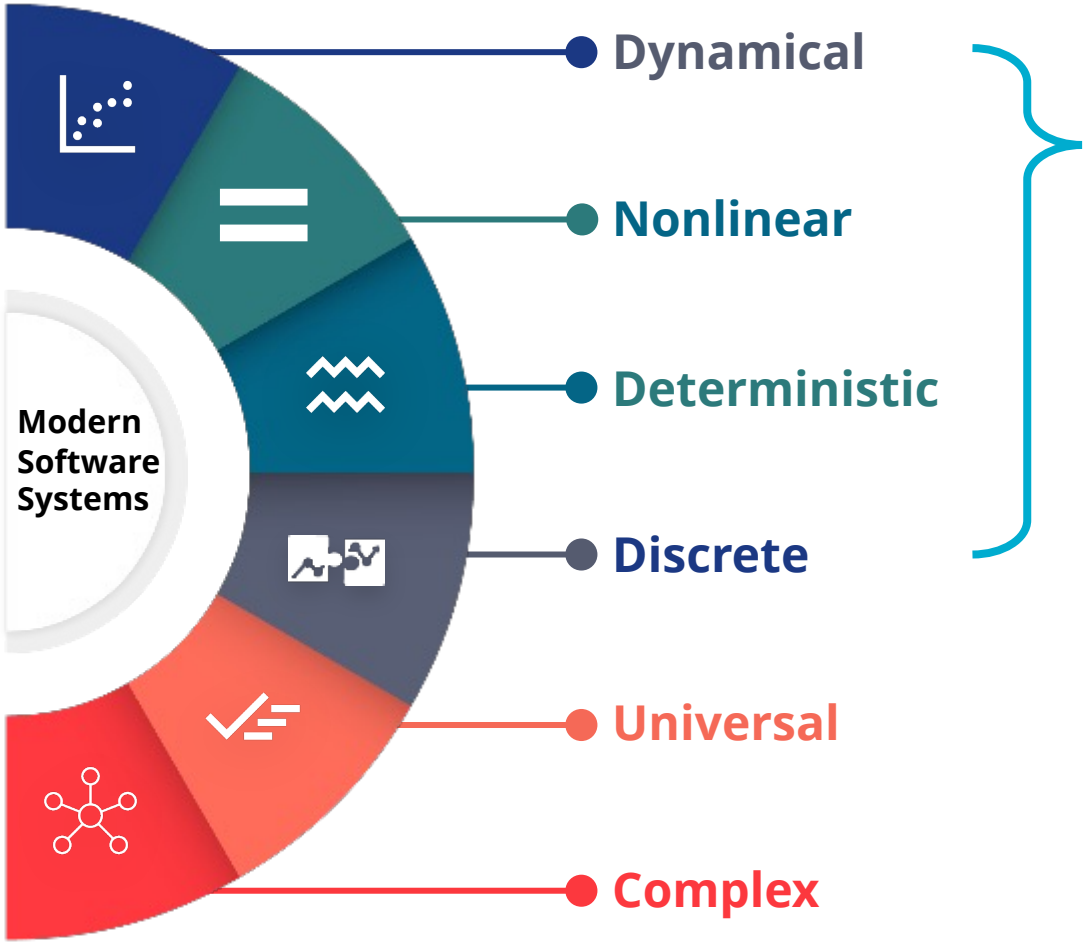
Also, these same characteristics are what makes software so effective in meeting functional requirements.

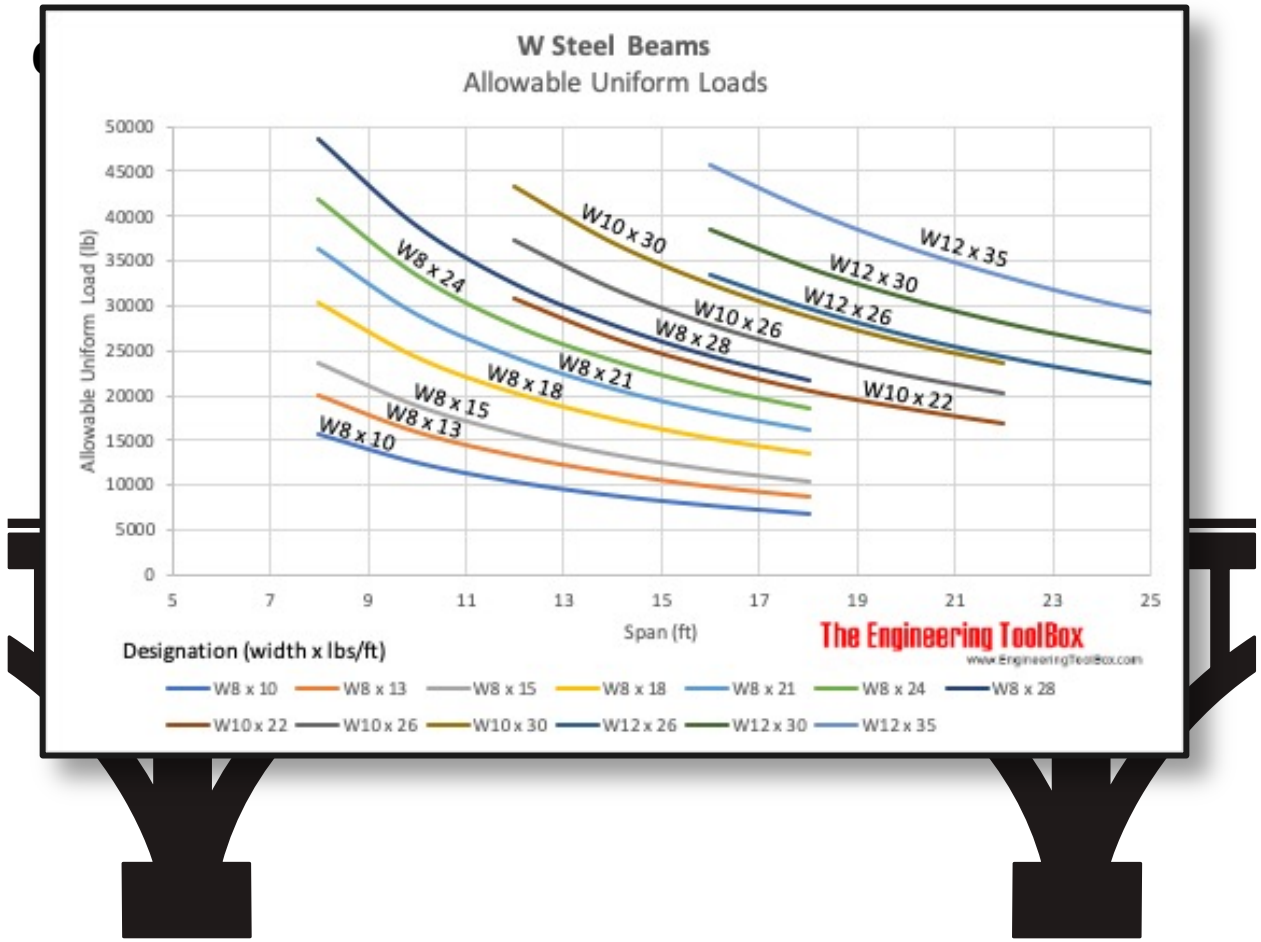
CHARACTERISTICS OF MODERN SOFTWARE SYSTEMS



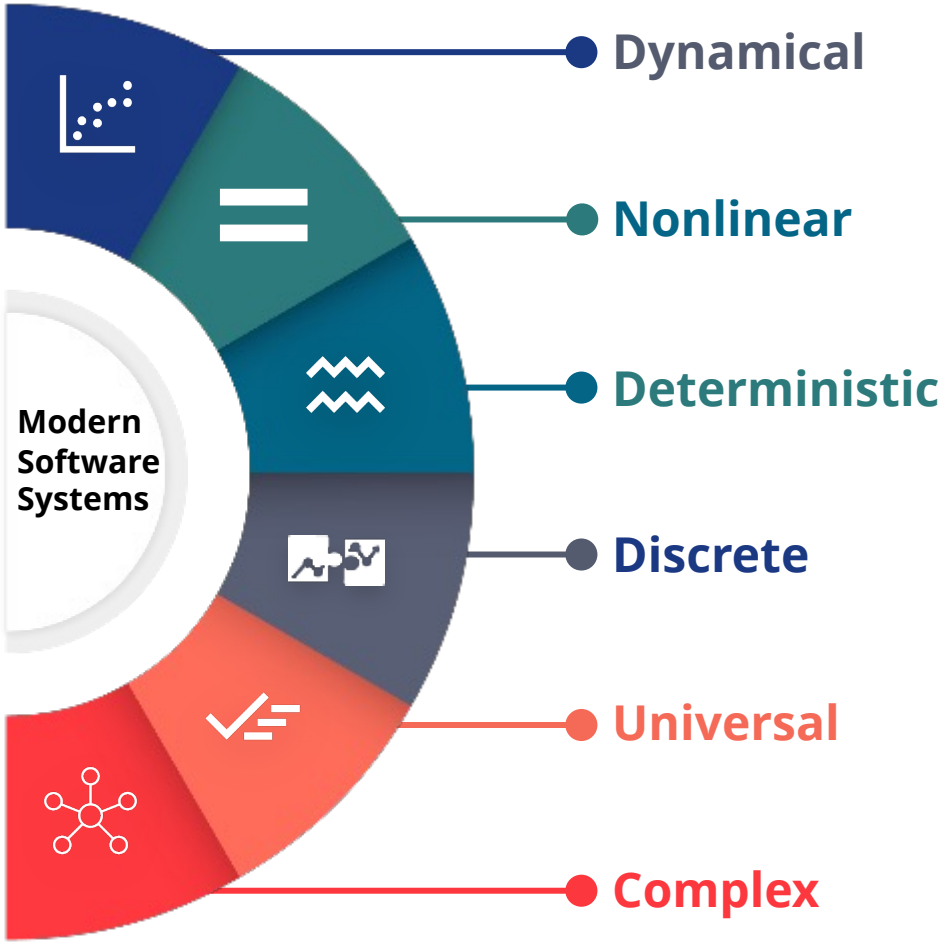
Our lives are steeped in continuous systems.

CHARACTERISTICS OF MODERN SOFTWARE SYSTEMS



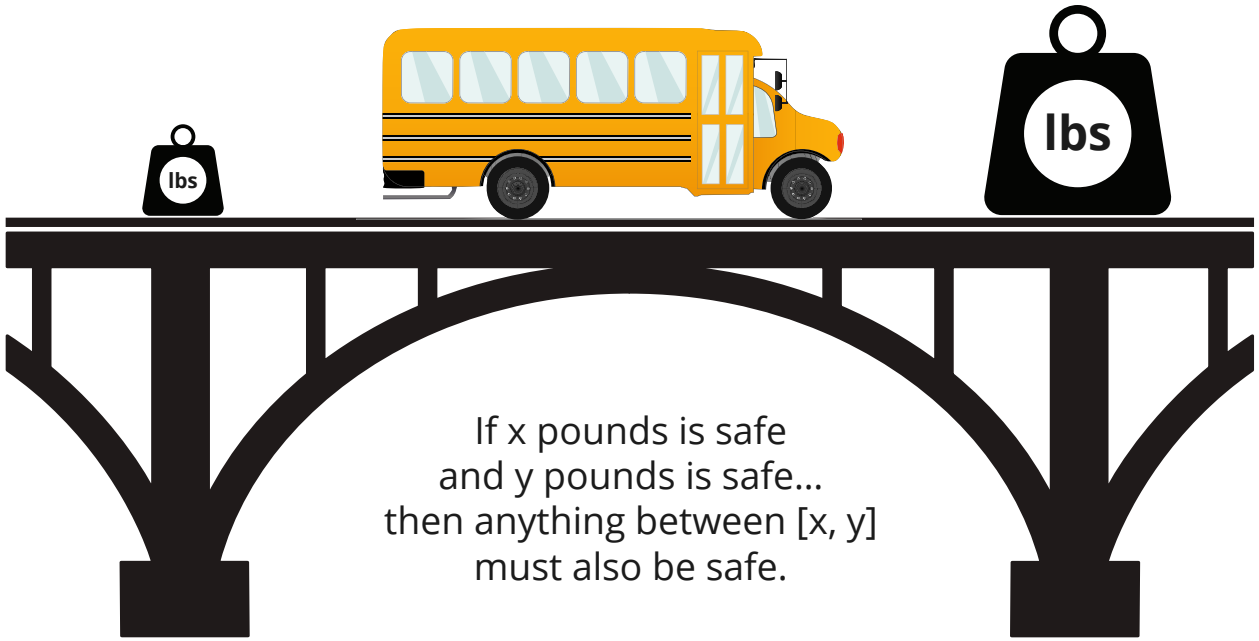


CHARACTERISTICS OF MODERN SOFTWARE SYSTEMS

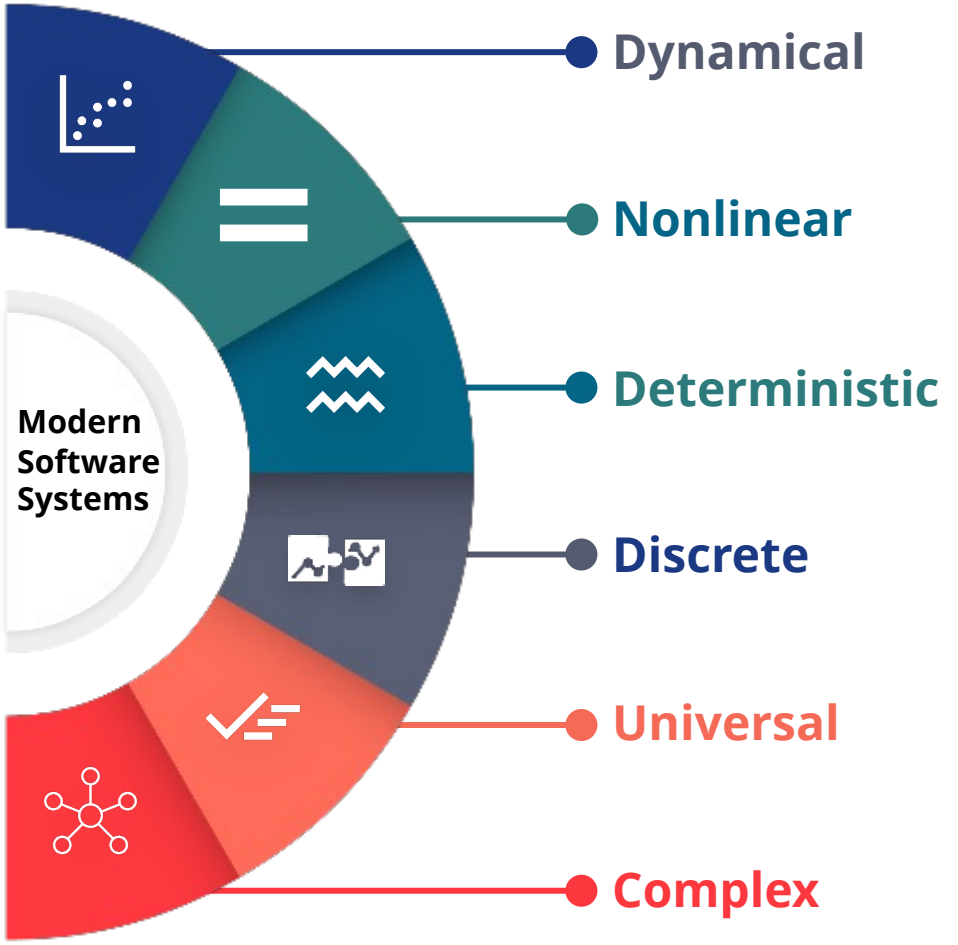


- Dynamical
- Nonlinear
- Deterministic
- Discrete
- Universal
- Complex

Our lives are steeped in continuous systems.



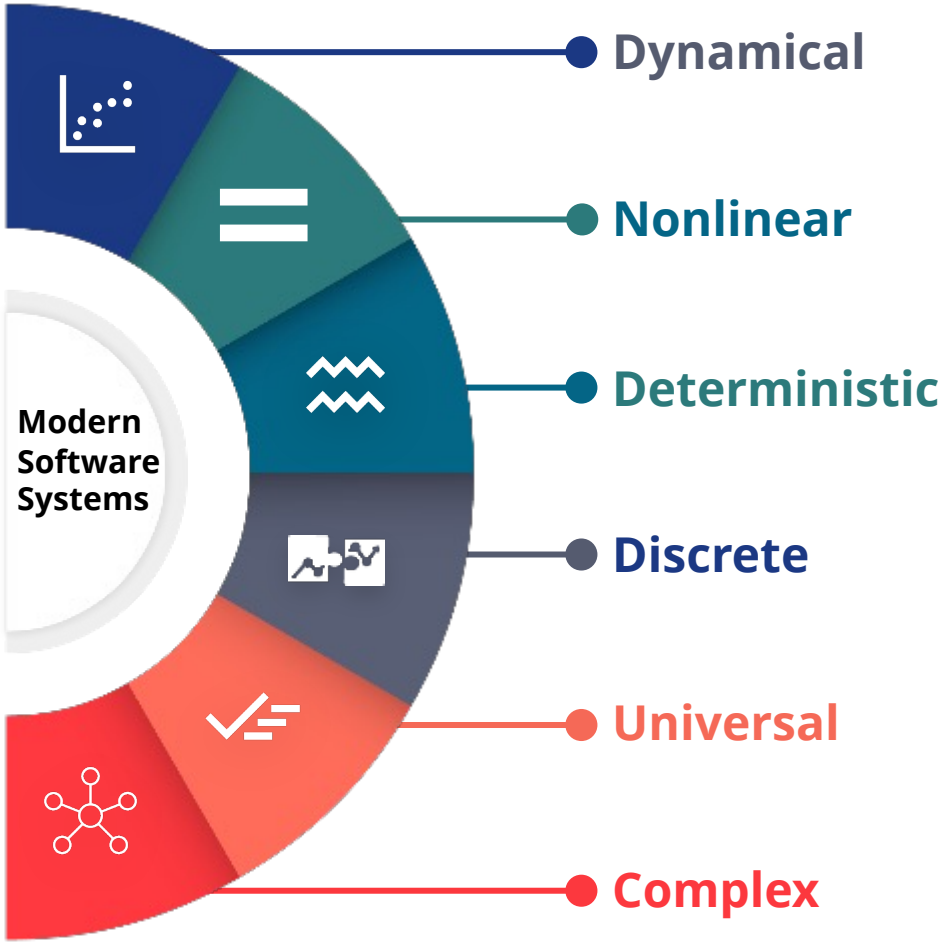
CHARACTERISTICS OF MODERN SOFTWARE SYSTEMS



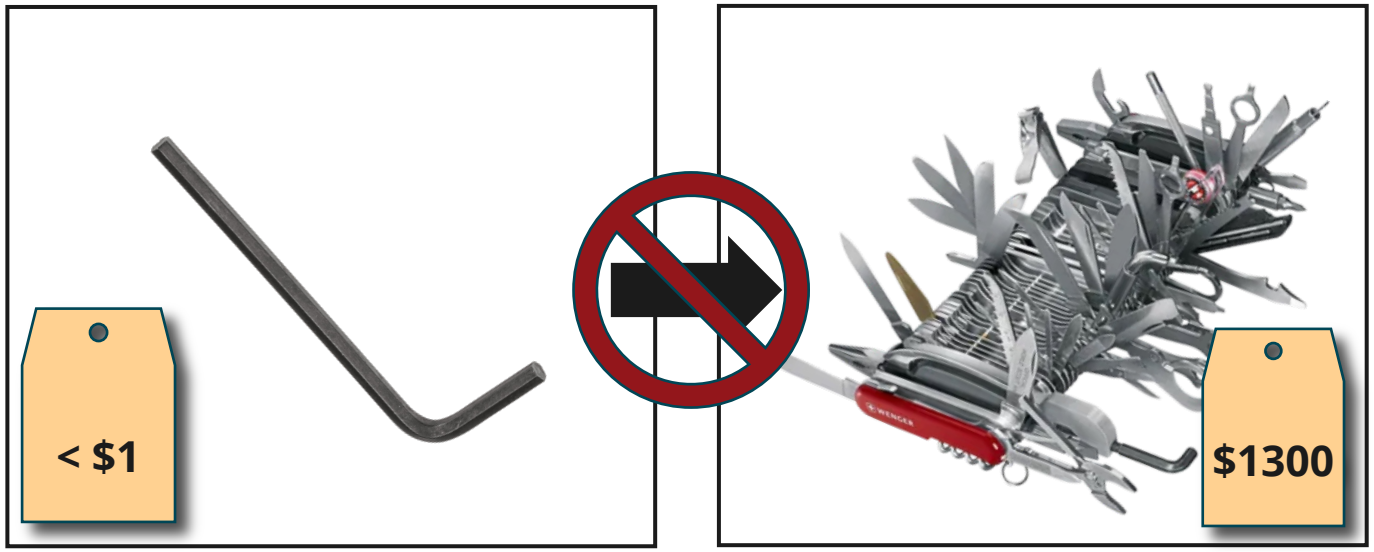
Software is discrete, not continuous.

Successfully testing software with inputs 2 and 4 tells you *nothing* about the behavior on input 3.

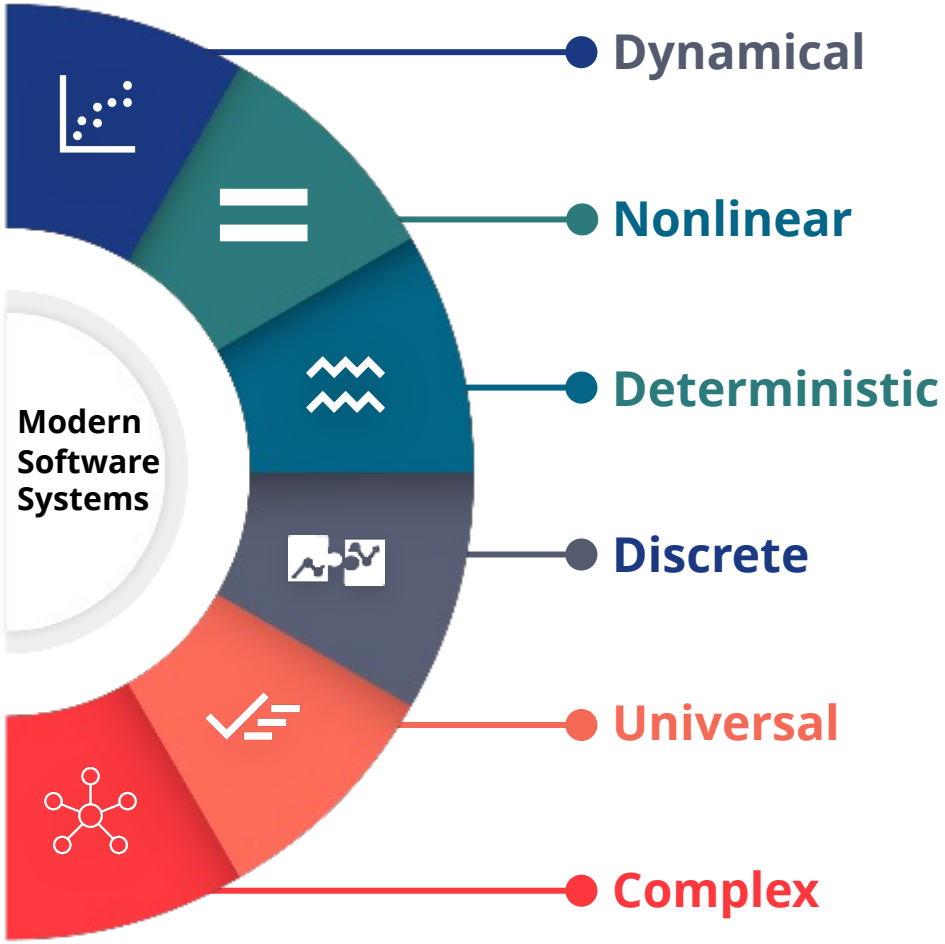
CHARACTERISTICS OF MODERN SOFTWARE SYSTEMS



We simulate simplicity with cheap complexity.



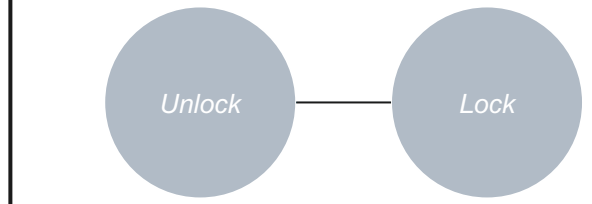
CHARACTERISTICS OF MODERN SOFTWARE SYSTEMS

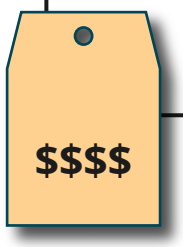


We simulate simplicity with cheap complexity.

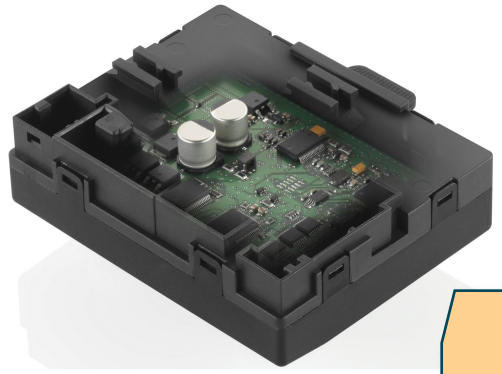
Car door locking systems should be simple.

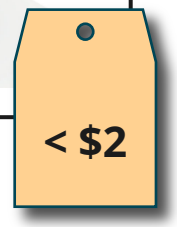
Simple but custom is expensive.



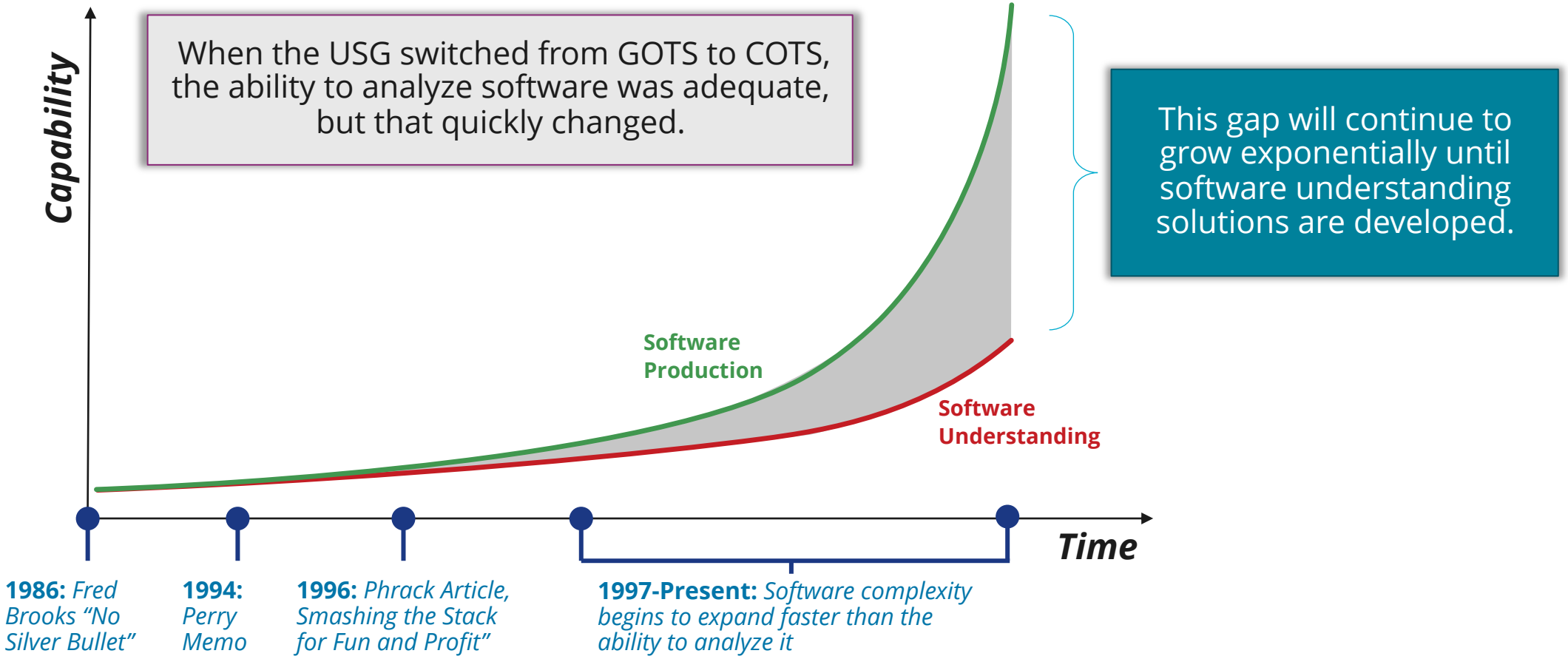


A Door Control Unit (DCU) has a full CPU, runs a real-time OS.





THE SOFTWARE UNDERSTANDING GAP



The software understanding gap is expanding exponentially.
The more it expands, the more it will impact national security and critical infrastructure missions.

INHERENT RISK VS. RESIDUAL RISK

Residual risk is the risk that remains after inherent risk has been partially mitigated.

Understanding inherent risk is the first step in the risk assessment process.

For software, the lack of adequate software understanding capability means that risk assessors cannot effectively implement the first step in the process, **rendering the rest of the process fundamentally flawed.**

HOW WE OPERATE TODAY

Today, an agency needing to analyze one piece of software to answer a mission question can fund an effort to do that analysis.

$$$

Could our sensitive data be changed without authorization?

Does this software have hidden ransomware?

Is there a kill switch hidden in my propulsion systems?

What indicators does this malware have?

Is there hidden, triggered behavior in our weapon's targeting systems?

Could sensitive communications be relayed off-ship?

Can my ship's location be tracked?

Is this software

Could our sens data be leaked

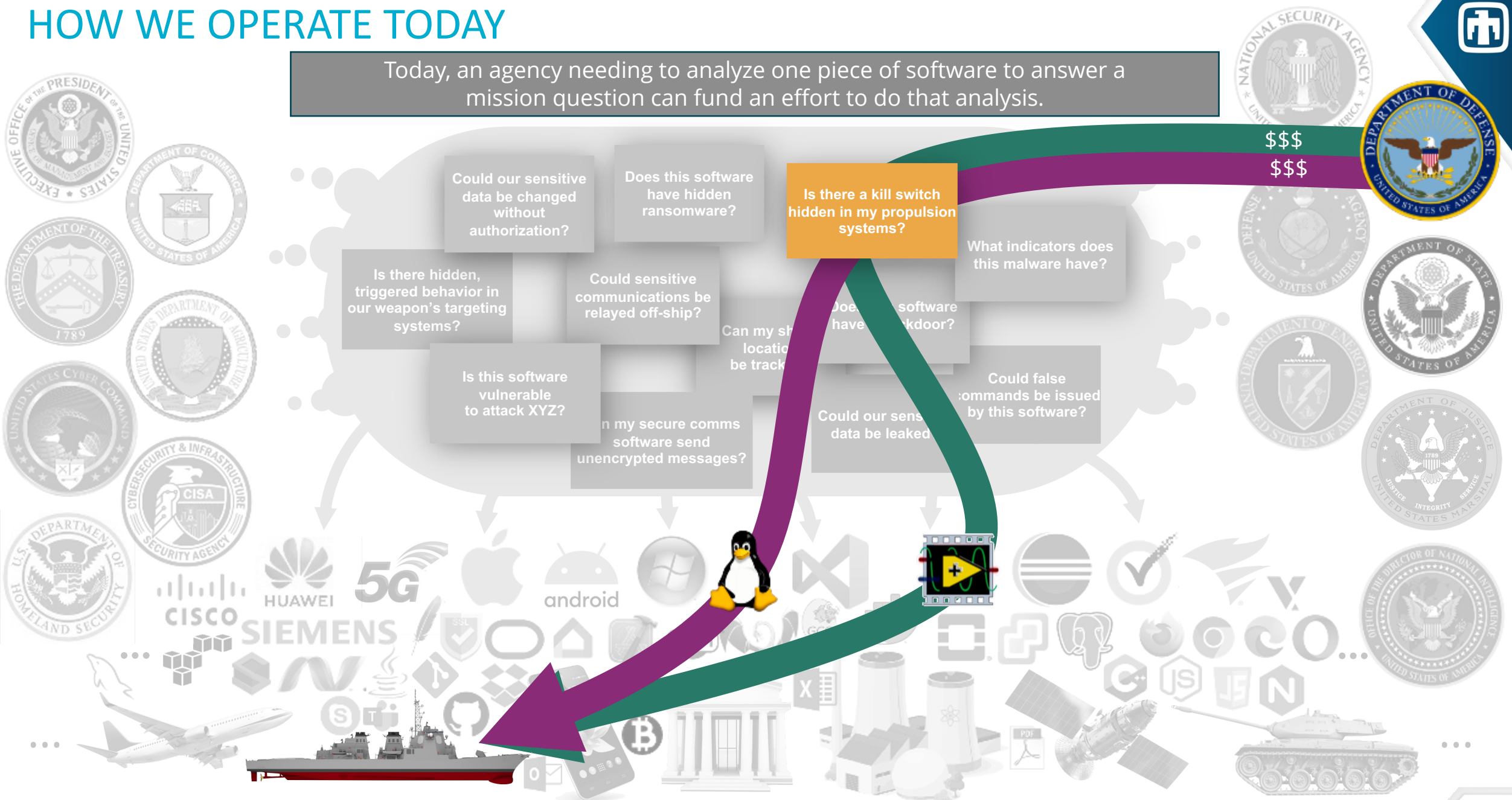
Could false commands be issued by this software?

This is a major undertaking.

Examples are entirely notional, for illustration purposes only.

HOW WE OPERATE TODAY

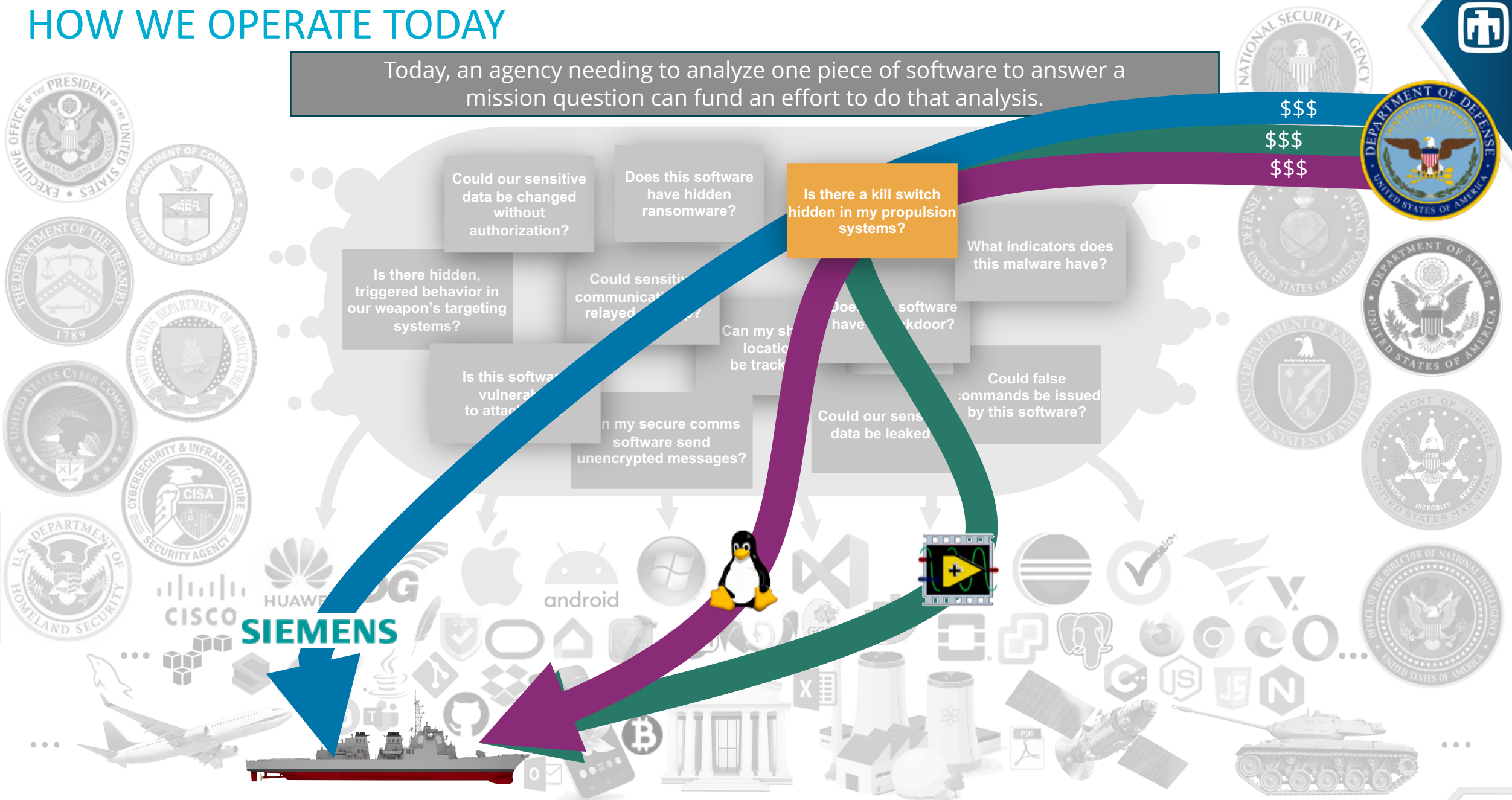
Today, an agency needing to analyze one piece of software to answer a mission question can fund an effort to do that analysis.



Examples are entirely notional, for illustration purposes only.

HOW WE OPERATE TODAY

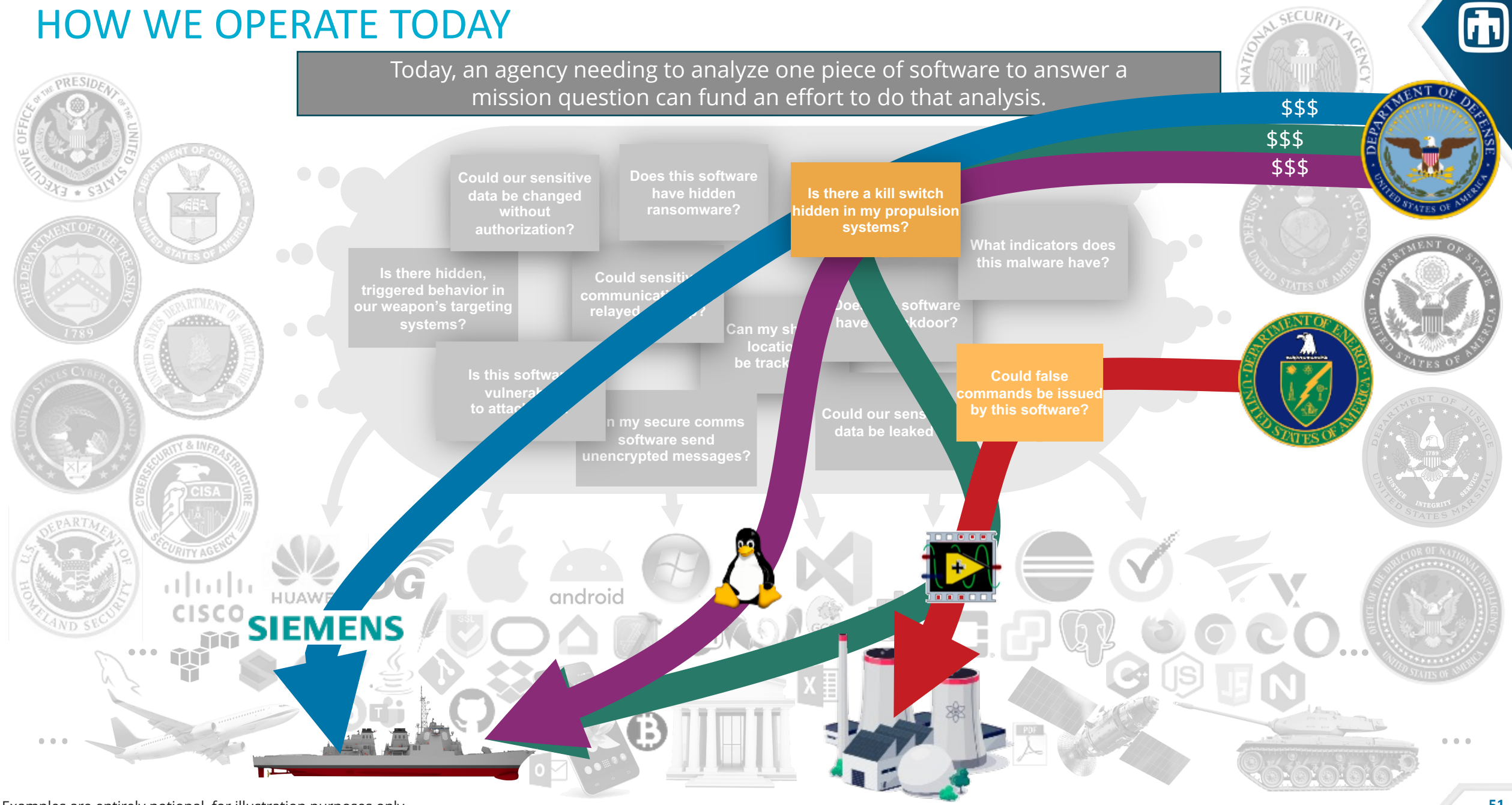
Today, an agency needing to analyze one piece of software to answer a mission question can fund an effort to do that analysis.



Examples are entirely notional, for illustration purposes only.

HOW WE OPERATE TODAY

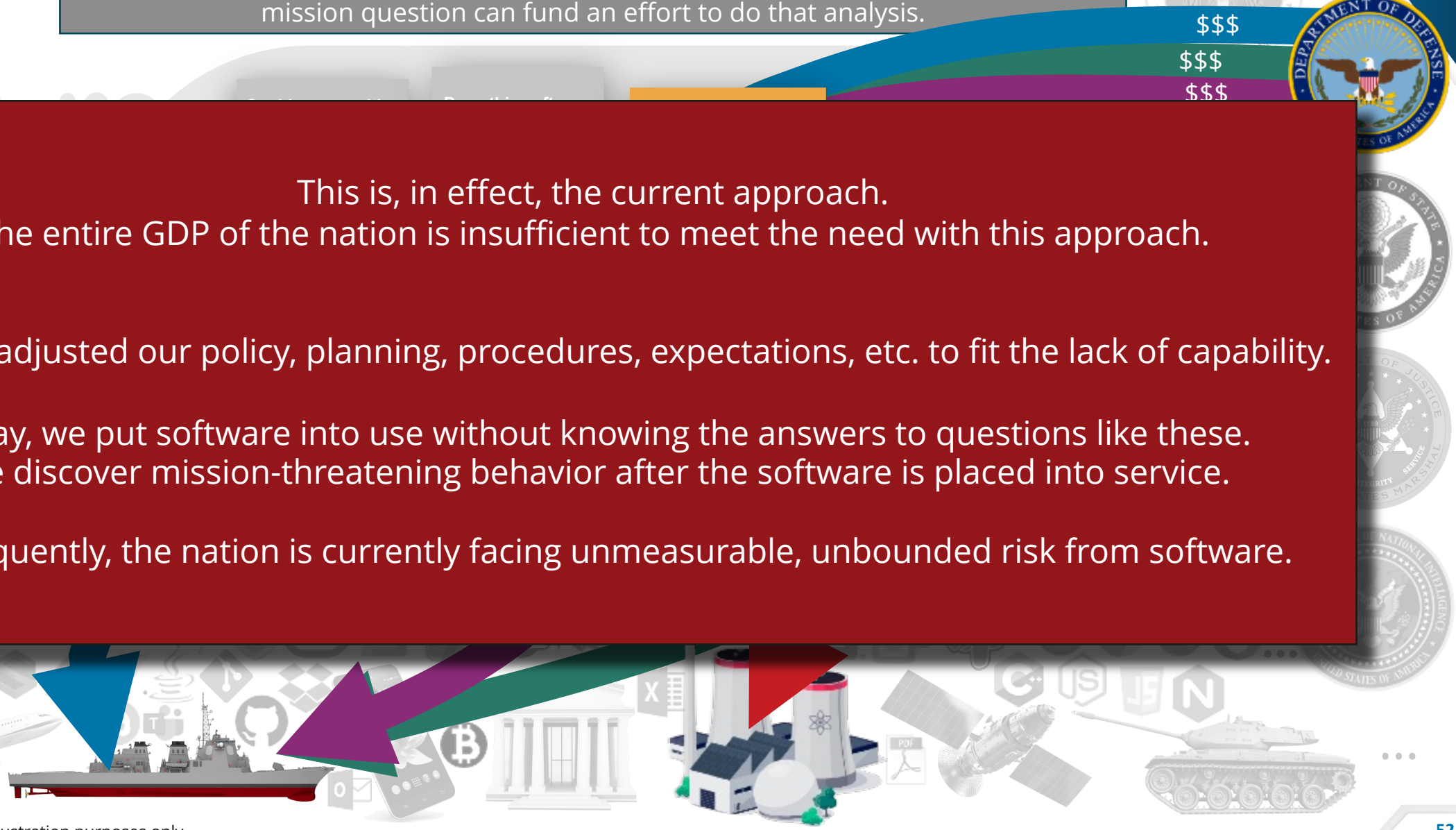
Today, an agency needing to analyze one piece of software to answer a mission question can fund an effort to do that analysis.



Examples are entirely notional, for illustration purposes only.

HOW WE OPERATE TODAY

Today, an agency needing to analyze one piece of software to answer a mission question can fund an effort to do that analysis.



This is, in effect, the current approach.
The entire GDP of the nation is insufficient to meet the need with this approach.

We have adjusted our policy, planning, procedures, expectations, etc. to fit the lack of capability.

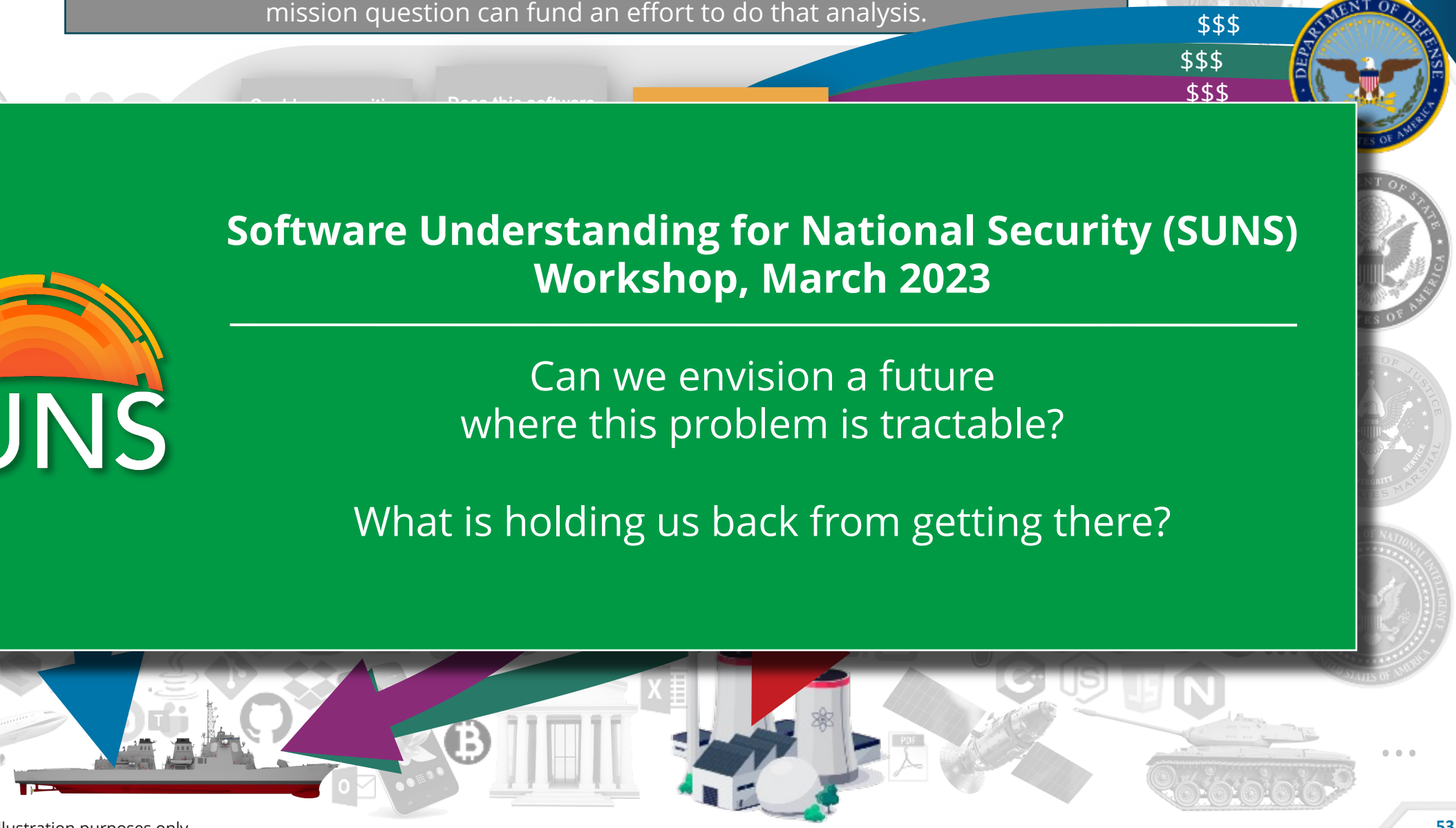
Today, we put software into use without knowing the answers to questions like these.
We discover mission-threatening behavior after the software is placed into service.

Consequently, the nation is currently facing unmeasurable, unbounded risk from software.

Examples are entirely notional, for illustration purposes only.

HOW WE OPERATE TODAY

Today, an agency needing to analyze one piece of software to answer a mission question can fund an effort to do that analysis.



Software Understanding for National Security (SUNS) Workshop, March 2023

SUNS

Can we envision a future
where this problem is tractable?

What is holding us back from getting there?

Examples are entirely notional, for illustration purposes only.

SUNS 2023 WORKSHOP

5 USG representatives co-convened ~30 technical SMEs from 10 USG research groups to discuss the need for a national capability for software understanding.

QUESTIONS PRESENTED

1. What are technical impediments to a national software understanding capability?
2. How can the USG support the expansion of software understanding?
3. What are near-term R&D funding priorities?

Top 5 Issues:

1. Lack of Unified Vision
2. Community Building
3. Lack of Sharing
4. Funding
5. Metrics and Benchmarks

Many expressed the opinion that a national cyber crisis is inevitable unless we revolutionize the way we analyze software.

SUNS 2023 WORKSHOP

Co-Conveners

CISA
Senior Technical Director for Cybersecurity Division

Associate Chief of Strategic Technology

NSA
Technical Director of Research

Technical Director of Cybersecurity

NNSA
Director, Nuclear Enterprise Assurance Division

Open Sessions

53 Attendees from:

Army/ESIC	MIT-LL	PNNL
CISA	NIST	SEI
DARPA	NSA	SNL
DHS/S&T	ODNI	ZRA
DIA	OUSD R&E	

Closed Sessions

29 Technical SMEs from:

DARPA	MIT-LL
CISA	NSA
IDA/CCS	PNNL
GTRI	SEI
LLNL	SNL

SUNS 2023 KEY RESULTS

10x-100x+ improvement in software understanding capabilities is possible, but progress is currently prevented by lack of a centralized vision, funding that is 10x+ too low, inability to collaborate, and other non-technical issues.

80% of SMEs expressed opinions consistent with this statement.

The scope of a national software understanding vision should include foundational research.

100% of SMEs expressed opinions consistent with this statement.

OUR CONCLUSIONS BASED ON THE SUNS 2023 DISCUSSIONS

#1 Radically improved technical capabilities for software understanding are possible.

#2 A unified national effort to revolutionize our software understanding capabilities is necessary to meet current and future mission needs.
We are far from being on track today.

#3 The nation that learns to best analyze and reason about software artifacts will dominate global geopolitics for the next century.

AN OBSERVATION ABOUT SOFTWARE ANALYSIS TOOLS

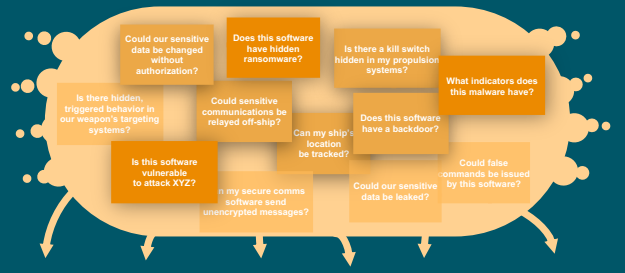
analyzer ≈ *f*(*mission_question*, *program_under_test*, *resource_tradeoffs*)

AN OBSERVATION ABOUT SOFTWARE ANALYSIS TOOLS

analyzer ≈ *f*(*mission_question*, *program_under_test*, *resource_tradeoffs*)

This term explains why a tool designed for one purpose is ill suited for others.

E.g., a tool designed to detect authentication backdoors is ill suited to identifying information leaks.



AN OBSERVATION ABOUT SOFTWARE ANALYSIS TOOLS

analyzer ≈ *f*(*mission_question*, *program_under_test*, *resource_tradeoffs*)

This term explains why tools don't work well across many programs.

E.g., what's needed to analyze real-time
flight controllers and web servers are different.

AN OBSERVATION ABOUT SOFTWARE ANALYSIS TOOLS

analyzer ≈ *f*(*mission_question*, *program_under_test*, *resource_tradeoffs*)

Varying resource and accuracy tradeoffs across mission applications is captured by this term.

E.g., national security missions may choose to invest far more computational resources than a typical laptop user.

AN OBSERVATION ABOUT SOFTWARE ANALYSIS TOOLS

analyzer ≈ *f*(*mission_question*, *program_under_test*, *resource_tradeoffs*)

When we build analysis tools without coordination, we need a different tool for each combination of question, program, and resource tradeoffs:

$$|analyzer| = |mission\ questions| * |programs| * |tradeoffs|$$

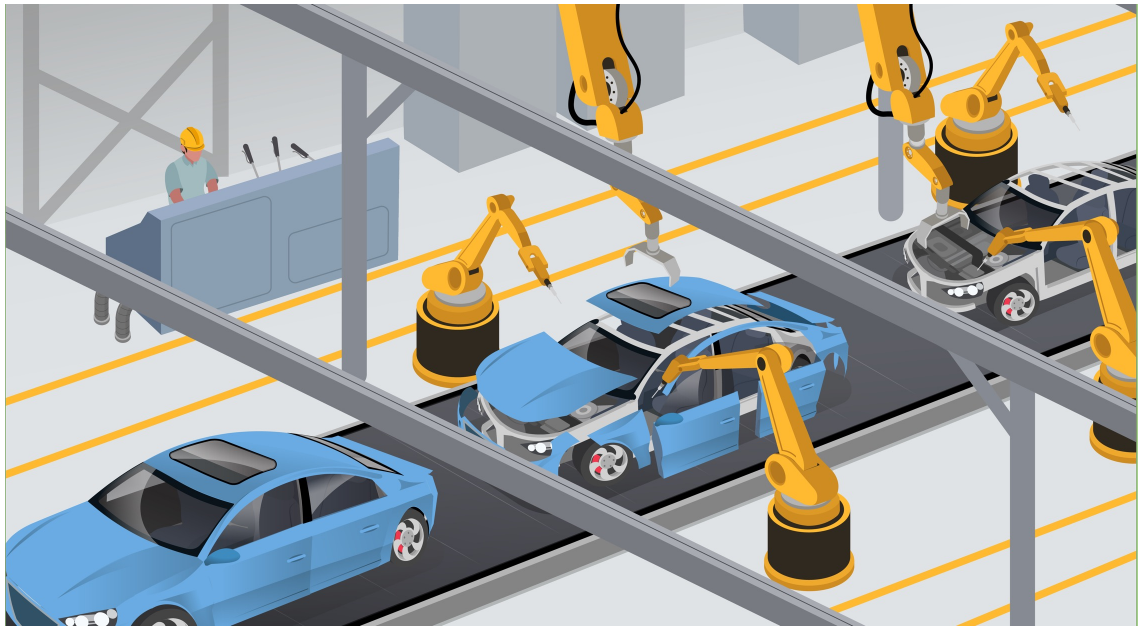
AN OBSERVATION ABOUT SOFTWARE ANALYSIS TOOLS

analyzer ≈ *f*(*mission_question*, *program_under_test*, *resource_tradeoffs*)

This is, in effect, the current approach.
The entire GDP of the nation is insufficient to meet mission needs
in software understanding using this approach.

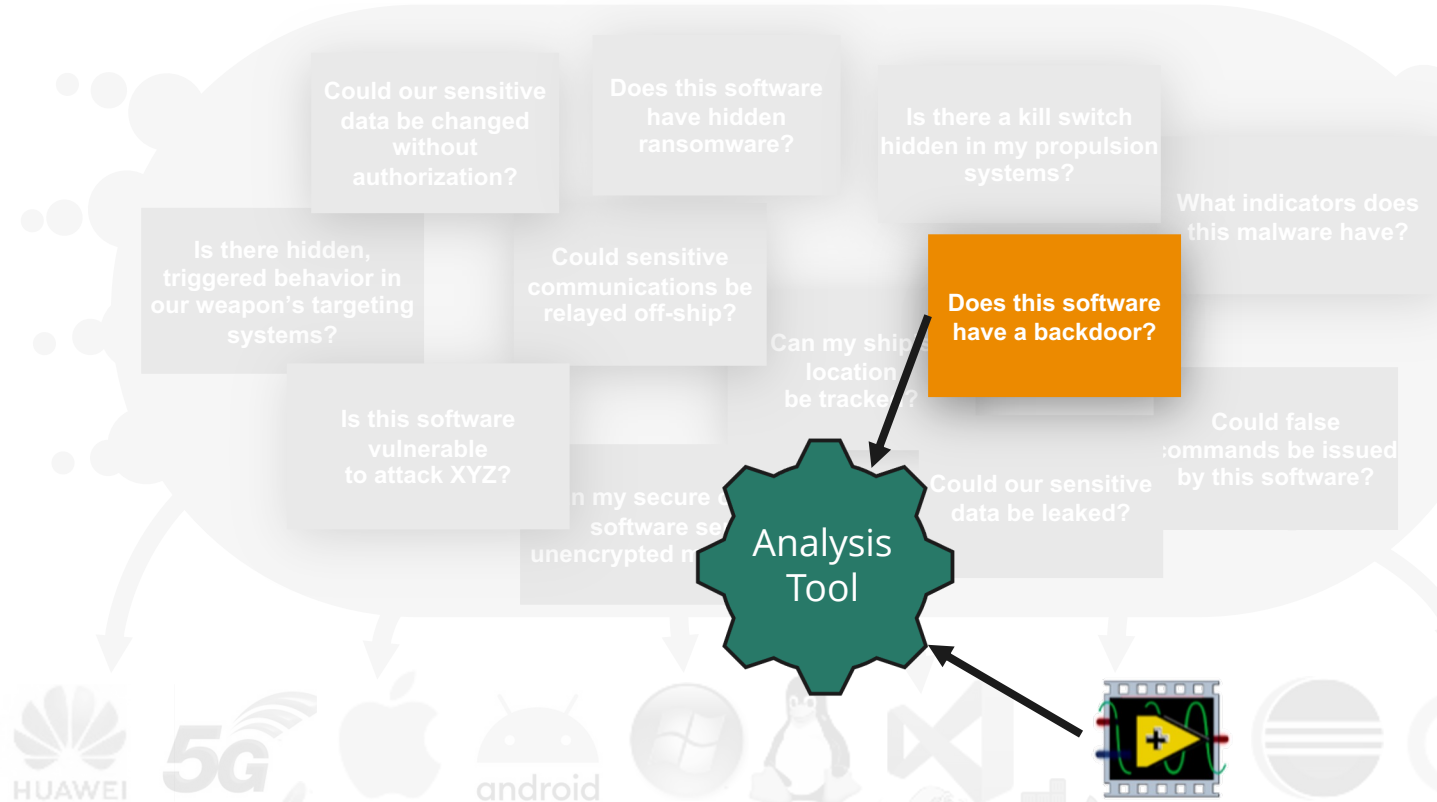
$$|analyzer| = |mission\ questions| * |programs| * |tradeoffs|$$

SCALABILITY REQUIRES A DIFFERENT APPROACH



- We need automated software understanding tools designed to scale across varying:
  - Mission Questions
  - Program executables
  - Resource tradeoffs

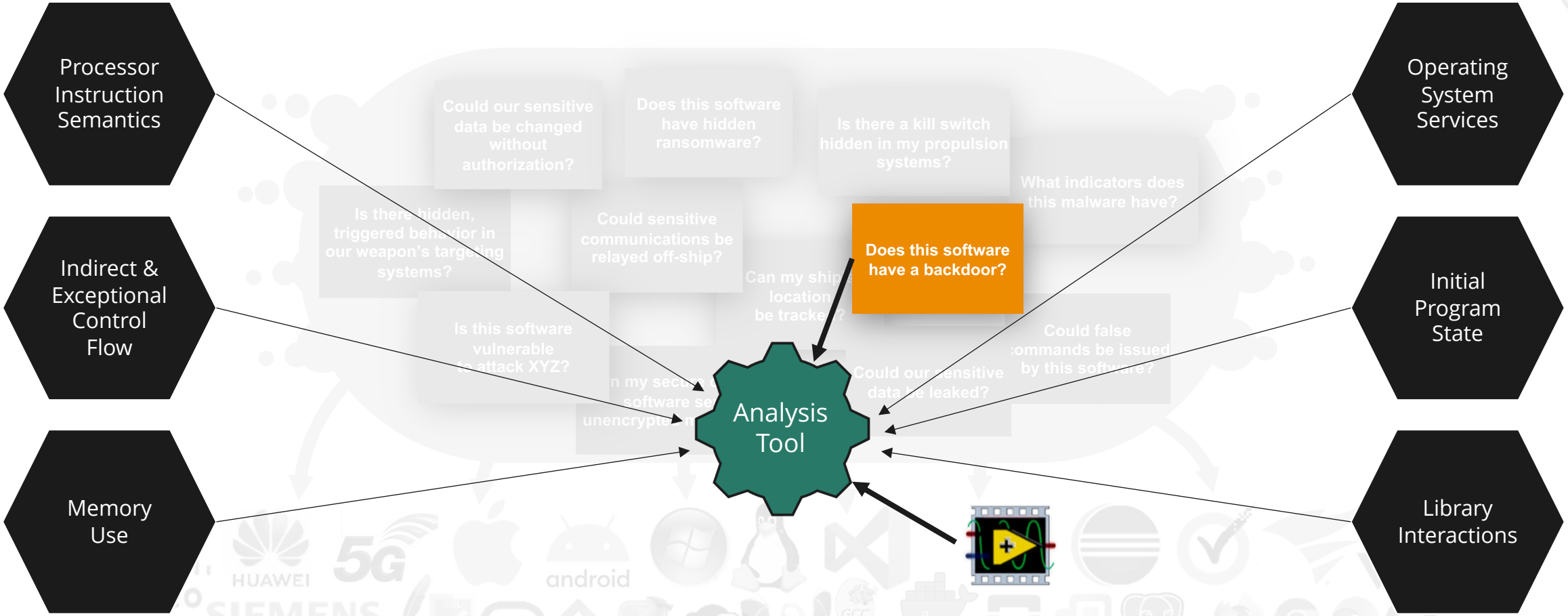
CONSIDER A TOOL FOR A SINGLE ANALYSIS TASK



To build an analysis tool to help answer a given question for a given program,
we need to model the execution of the program in order to analyze its potential behavior.

Examples are entirely notional, for illustration purposes only.

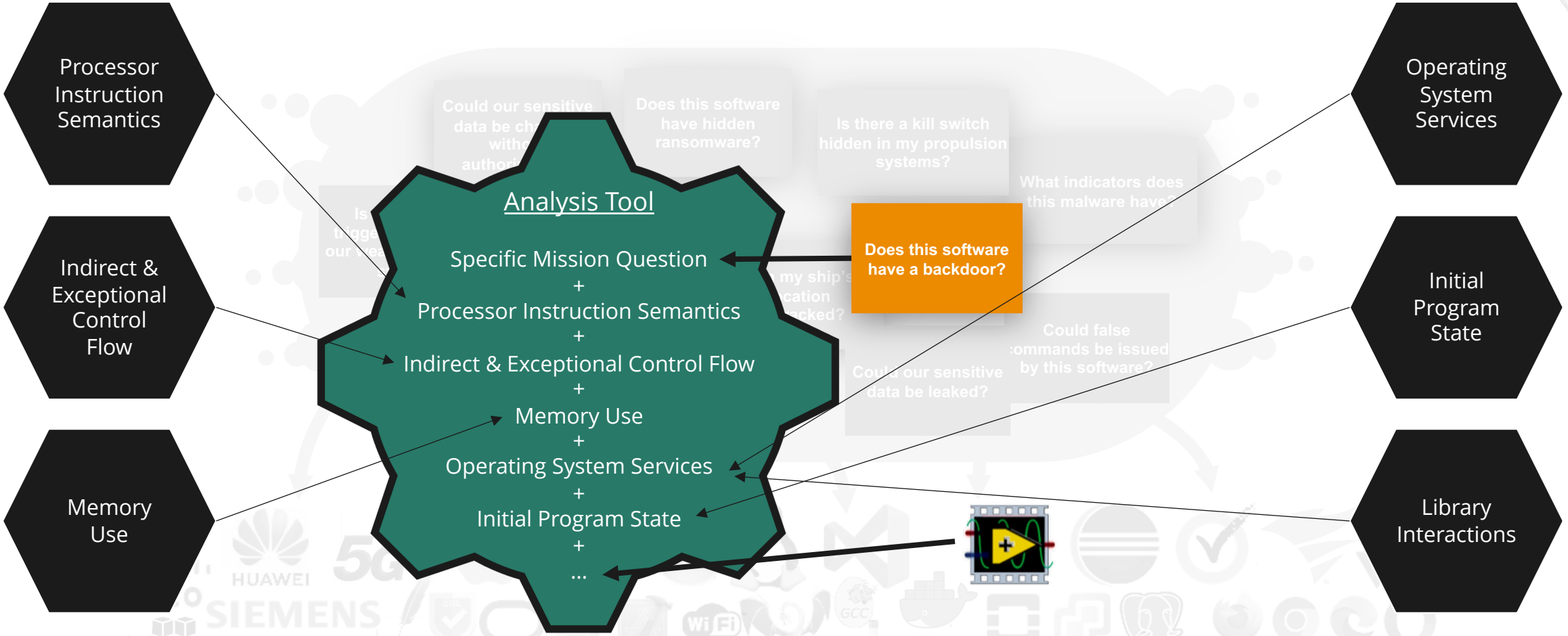
CONSIDER A TOOL FOR A SINGLE ANALYSIS TASK



To build an analysis tool to help answer a given question for a given program,
we need to model the execution of the program in order to analyze its potential behavior.

Examples are entirely notional, for illustration purposes only.

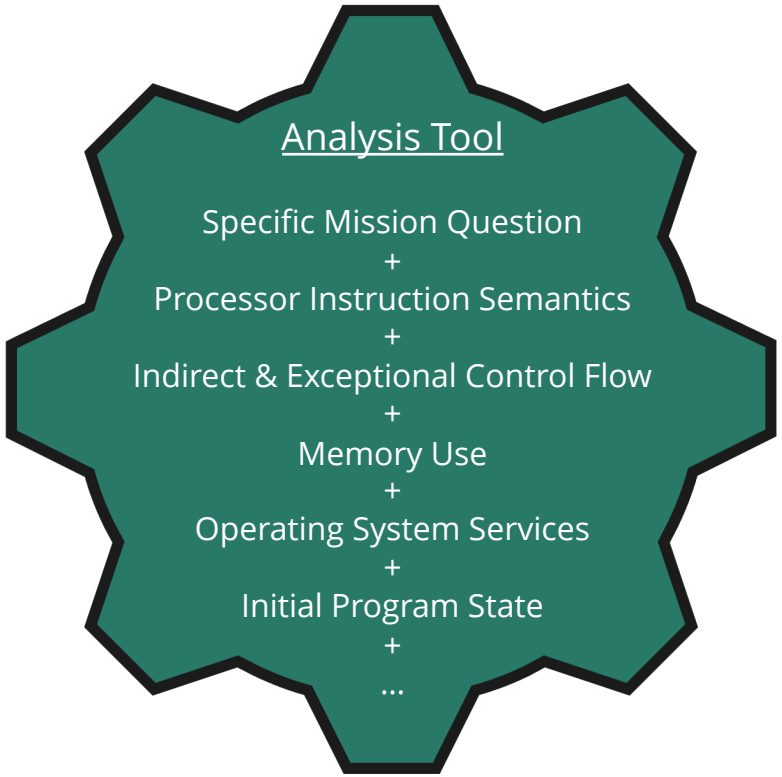
CONSIDER A TOOL FOR A SINGLE ANALYSIS TASK



Most tools today are largely monolithic, hard-coding various modeling decisions.

Examples are entirely notional, for illustration purposes only.

OBSERVATIONS ABOUT MONOLITHIC ANALYSIS TOOLS



Observation #1

- Poor models mean poor analysis results
- There is no single "right" answer for:
  - All mission questions
  - All programs
  - All customer needs, resources, and risk appetite

Monolithic, bespoke implementations are not generally reusable.

Observation #2

- Each analysis tool is a "chain link problem"
- It's only as good as its weakest link

Foundational, principled approaches for reusable components are disincentivized.

SCALABILITY DEMANDS DIFFERENT APPROACHES

Processor Instruction Semantics

Operating System Services

Indirect & Exceptional Control Flow

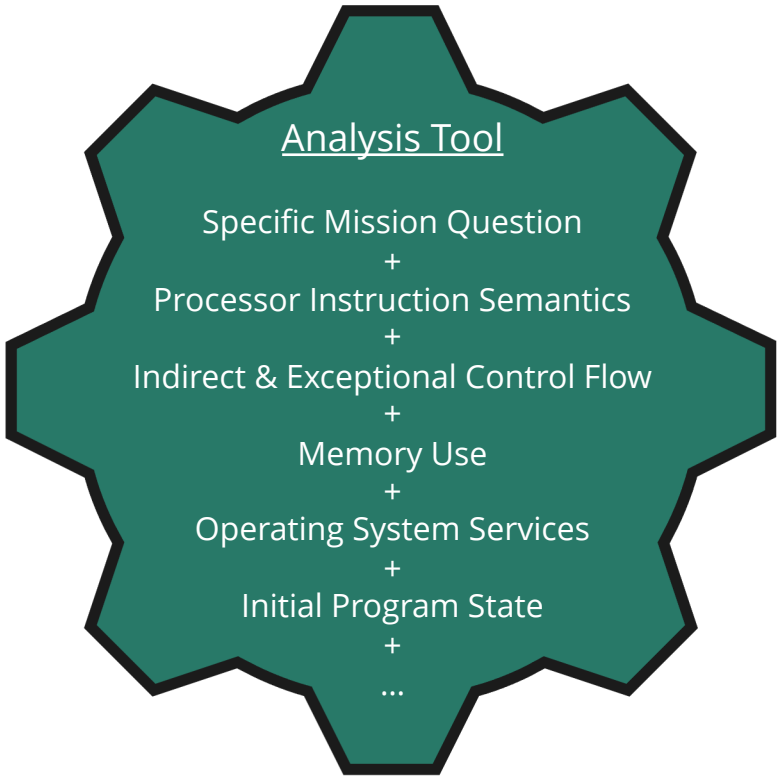
Initial Program State

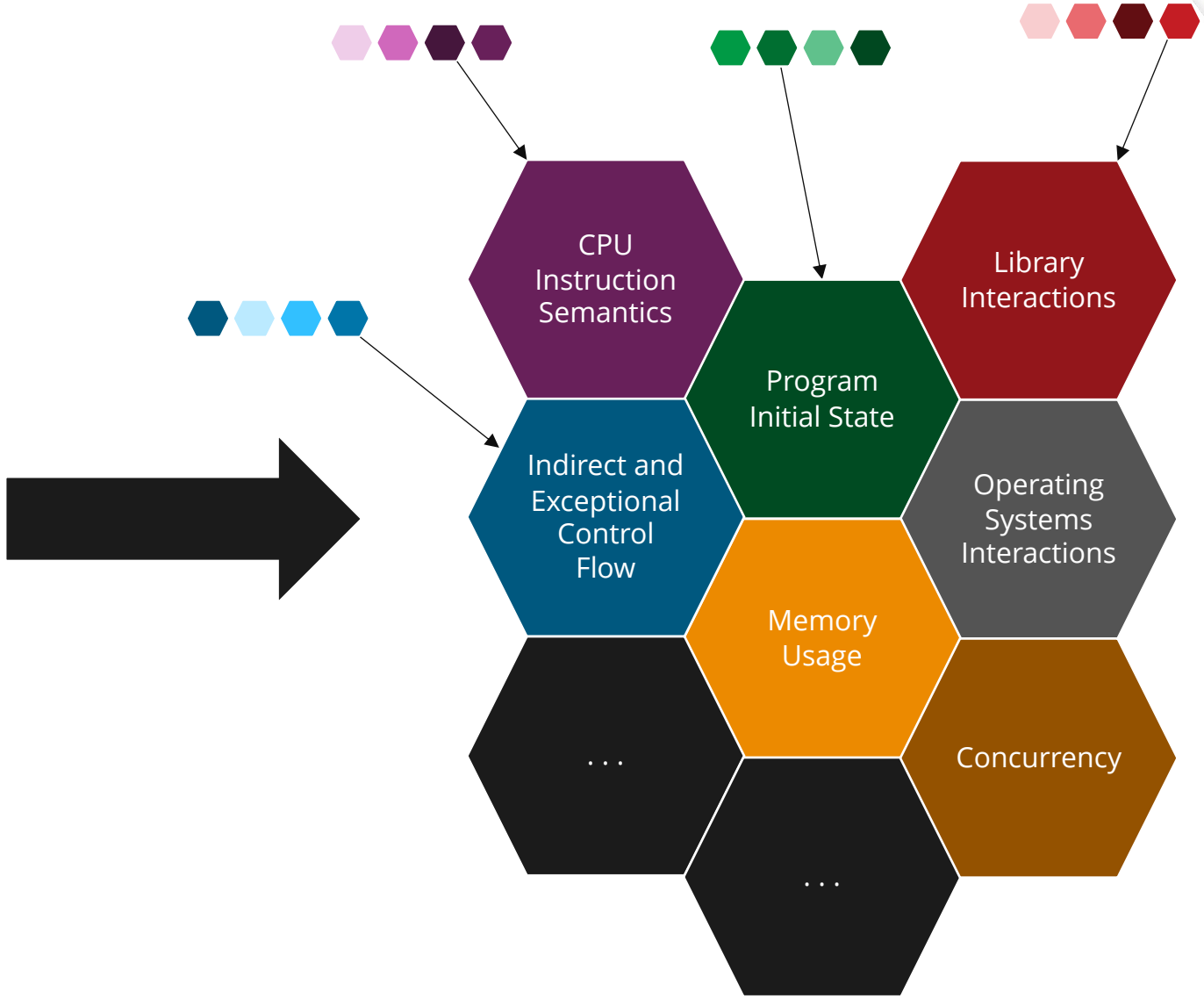
Memory Use

Library Interactions

Examples are entirely notional, for illustration purposes only.

SCALABILITY DEMANDS DIFFERENT APPROACHES

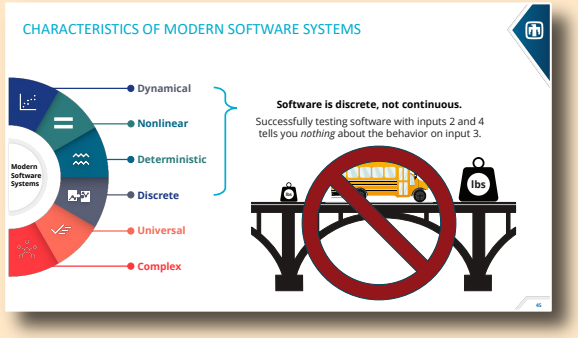


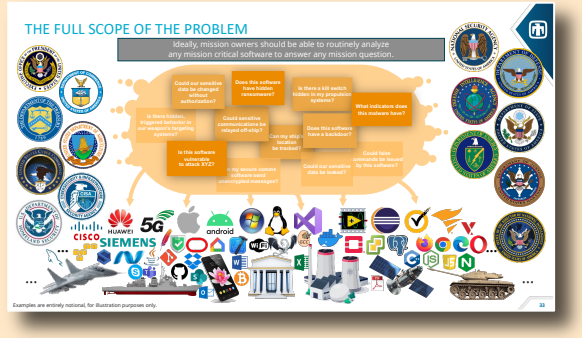


The current USG R&D landscape
is pushing the community the wrong way.

SCALABILITY DEMANDS DIFFERENT APPROACHES

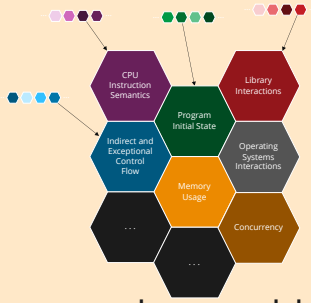
Technical Challenges





Cross-Agency Span

This challenge spans government.



Solution approaches could span agencies with a coordinated effort.

No department or agency has the charter or ability to explore solutions that scale to the entire USG.

CREATING A NATIONAL SOFTWARE UNDERSTANDING CAPABILITY

Recommendation #1: Make a national decision to address the software understanding gap

- The White House must direct coordination across Departments and Agencies
  - Whole of government & society effort
  - Senior technical SMEs directing investments
  - Continuity of effort across the R&D community

Recommendation #2: Create a cross-agency Software Understanding for National Security Oversight Council (SUNSOC)

- Provide national coordination across community engagements
- Establish collaborative agreements and environments
- Provide technical SME direction by developing and maintain a national R&D roadmap
- Manage national investments in software understanding
- Cultivate the software understanding community as a national resource

QUESTIONS?