Building Analytics for Network Flow Records

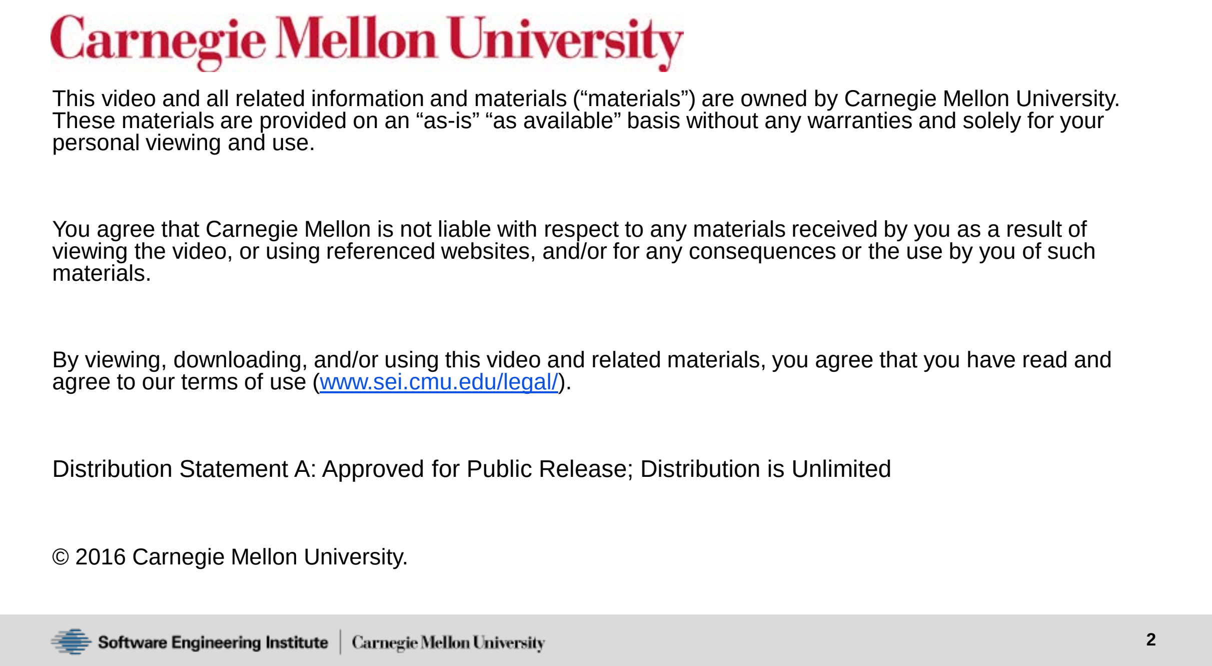
Table of Contents

SEI WEBINAR SERIEW | Keeping you informed of the latest solutions

SEI WEBINAR SERIES | Keeping you informed of the latest solutions

Software Engineering Institute | Carnegie Mellon University

Carnegie Mellon University



Carnegie Mellon University

This video and all related information and materials (“materials”) are owned by Carnegie Mellon University. These materials are provided on an “as-is” “as available” basis without any warranties and solely for your personal viewing and use.

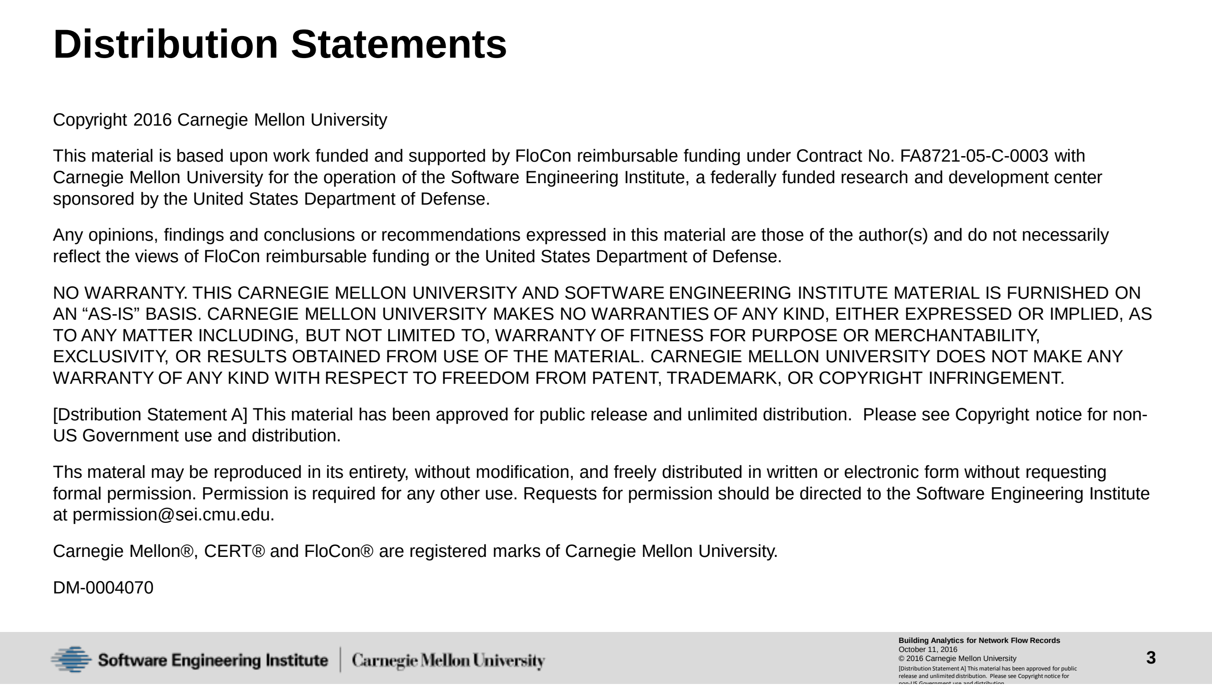
You agree that Carnegie Mellon is not liable with respect to any materials received by you as a result of viewing the video, or using referenced websites, and/or for any consequences or the use by you of such materials.

By viewing, downloading, and/or using this video and related materials, you agree that you have read and agree to our terms of use (www.sei.cmu.edu/legal/).

Distribution Statement A: Approved for Public Release; Distribution is Unlimited

Software Engineering Institute | Carnegie Mellon University 2

Distribution Statements



Distribution Statements

Copyright 2016 Carnegie Mellon University

This material is based upon work funded and supported by FloCon reimbursable funding under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center sponsored by the United States Department of Defense.

Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of FloCon reimbursable funding or the United States Department of Defense.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN “AS-IS” BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

[Dstribution Statement A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

Ths materal may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

Carnegie Mellon®, CERT® and FloCon® are registered marks of Carnegie Mellon University.

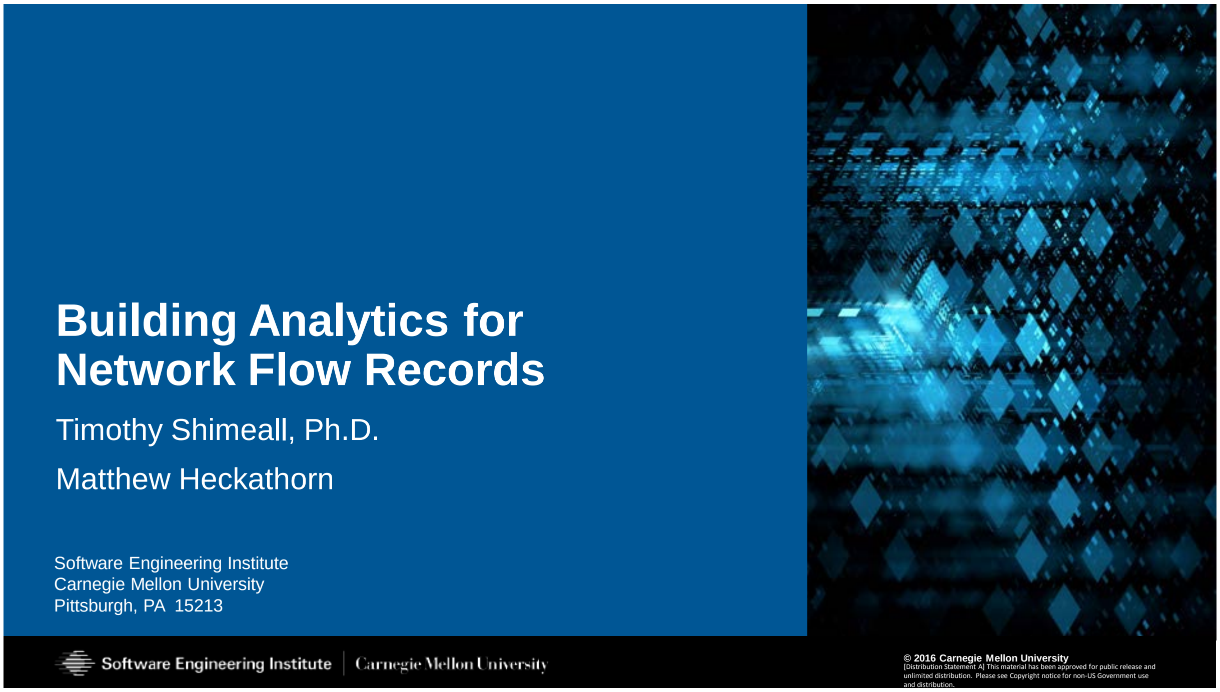
DM-0004070

Software Engineering Institute | Carnegie Mellon University

Building Analytics for Network Flow Records
October 11, 2016
© 2016 Carnegie Mellon University
[Distribution Statement A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

3

Building Analytics for Network Flow Records



Building Analytics for Network Flow Records

Timothy Shimeall, Ph.D.

Matthew Heckathorn

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Software Engineering Institute | Carnegie Mellon University

© 2016 Carnegie Mellon University
[Distribution Statement A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

**004 Moderator: And hello from the campus of Carnegie Mellon

University in Pittsburgh, Pennsylvania. We welcome you to the Software Engineering Institute's webinar series. Our presentation today is "Building Analytics for Network Flow Records." Depending on your location, we wish you a good morning, a good afternoon or good evening.

My name is Shane McGraw. I'll be your moderator for the presentation, and I'd like to thank you for attending. We want to make today as interactive as possible, so we will address questions throughout the presentation and again at the end of the presentation. You can submit your questions to our event staff at any time by using the Question or the Chat tabs on your Control Panel. We will ask a few polling questions throughout the presentation and they will appear as a pop-up window on your screen. The first question we'd like to ask is how did you hear about today's event?

Another three tabs I'd like to point out are the Download Materials, Twitter and Survey tabs. The Download Materials tab has a PDF copy of today's presentation slides there now, along with other network security-related work from the SEI. For those of you using Twitter, be sure to follow @CERT_Division, and use the #seiwebinar. And lastly, our Survey tab we ask that you fill out upon exiting today's webinar, as your feedback is always greatly appreciated.

Now I'd like to introduce our presenters for today. Timothy J. Shimeall is a Senior Member of the Technical Staff at the SEI, serving with the CERT Network Situational Awareness Group. He's responsible for the development of methods to support decision-making in cybersecurity at and above the enterprise level, including identification of useful data and analysis methods.

Matt Heckathorn is a Network Security Analyst at the CERT Division. He is a member of the Operational Analysis Transition team. He is tasked with the raising of awareness of the tools and knowledge the CERT Division can provide. He co-authored the SEI technical report "Network Monitoring for Web-Based Threats," which provides guidance to network analysts in identifying and defending against web-based threats.

So Matt, Tim, welcome. All yours.

Instructor: Glad to be here.

Instructor: Thanks, Shane.

Instructor: So Matt, I understand that you've done several of these webinars?

Instructor: Ah, no. Actually, this is my first. I've been at the SEI for about five years now, and I think that this is a great service that we offer. How about you, Tim?

Instructor: This is actually my first too. I'm anxious to see how well it works and get some interaction with the folks. I know I've worked with you a bunch on a variety of different projects, including the SiLK "Analyst's Handbook."

Instructor: Yep.

Instructor: And that provides for some interesting stuff. But we really want to be kind of responsive to the audience today, and part of what we'd like to do is get a feel for what the maturity point is of the audience with respect to their use of network situation awareness. So I think that leads us into our first poll.

Instructor: Yeah.

Moderator: Okay. So you're going to see that pop up on your screen now. And again, we're going to ask a couple polling questions throughout the presentation and it's going to help drive the flow of the presentation for examples for Matt and Tim to take the presentation. So the question is, "How is your organization automating analysis for network situational awareness for network security?" So we'll give you about 15 or 20 seconds to vote on that. There's quite a few options. We'll let you guys continue.

Instructor: Yeah. So I think the intent behind the first question is to try to tailor our talk a little bit better today. There's quite a bit of content that we could go over. It's a fairly long--the process will vary based on how mature an organization actually is when it comes to network flow analytics.

Instructor: Certainly. I don't think we wanted to jump in and just assume that there's expertise or--and so confusion. On the other hand, if there is a lot of expertise already, then we want, we can move, more rapidly over some of the more basic material.

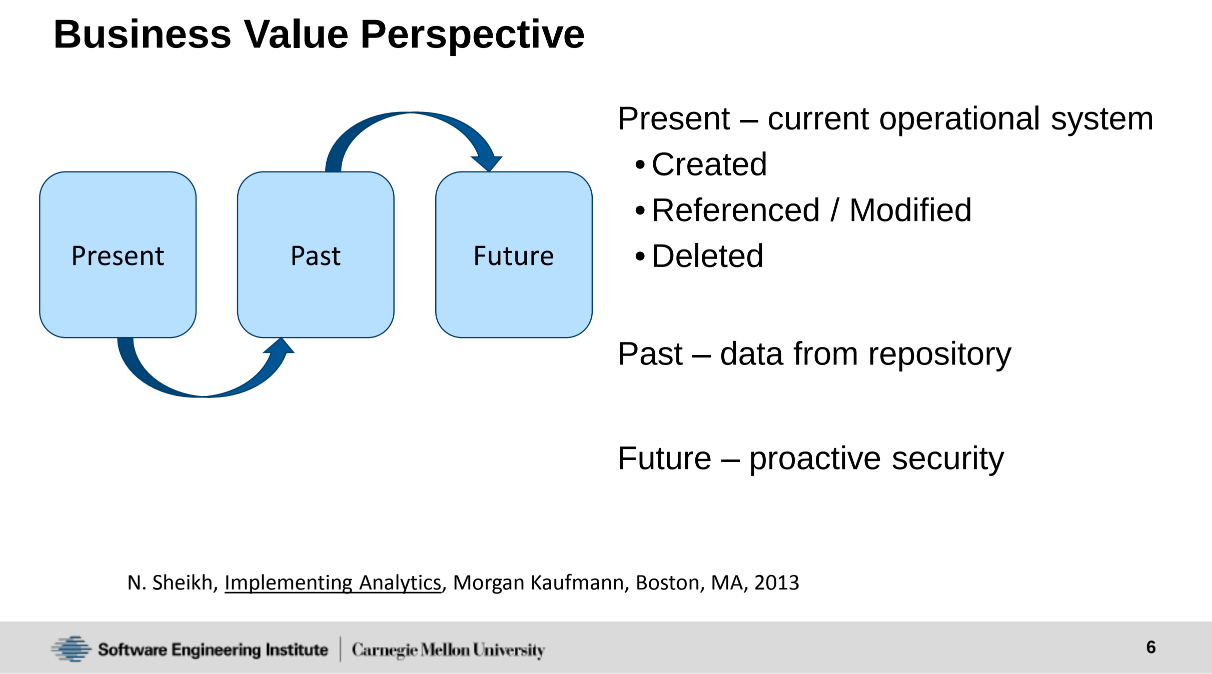
Instructor: Yeah. I think hopefully we can get some good answers here then.

Moderator: Okay. So we'll look at those results now. We got 21 percent that are uncertain; 26 percent heroes; 11 percent repeatable; and 21 percent optimized. We have a process and a focus, continuing to improve it. So little bit, little bit of a mix of results for everything.

Instructor: Yeah. I think fairly mixed across the arrays of, you know, maturity here. Right. I think with heroes, you know, we're probably looking to maybe front-load a little bit more on some things, but with optimized, back load. So we have a nice, nice spread.

Instructor: Yeah. I think that's a good, good group. And certainly if we start to, to bring things together and we pass something that one of the attendees doesn't understand, we really hope that they'll ask questions.

Business Value Perspective



Business Value Perspective

Present – current operational system

- Created
- Referenced / Modified
- Deleted

Past – data from repository

Future – proactive security

N. Sheikh, Implementing Analytics, Morgan Kaufmann, Boston, MA, 2013

Software Engineering Institute | Carnegie Mellon University 6

**006 Moderator: And I'm going to chime in real quick. Just one of the options we did, and I forgot to mention this, was "Other." And we want them to type into the chat. Jamel chimed in with "Balanced, of out-sourced, in-house, and product-based solutions."

Instructor: Okay. So across a couple answers I think is what the indication there is. In general, it seems like, you know, we have sort of a wide degree, which I don't think is completely off the mark of what we were thinking we would end up having to do.

Moderator: Okay.

Instructor: So...

Instructor: Okay.

Instructor: Okay. All right. Well, I think we'll jump into the presentation then. So I think one of the first things that we kind of wanted to do here was set expectations a little bit. We mentioned SiLK. And in particular, we wanted to come at this from something that some people might be aware of, which is sort of a business value perspective. So the ideas that analytics can range across both past, present and future, with present being things like operations systems, you know, near real-time. Things like things that have been created or referenced or modified or deleted recently. And then there's the past, right. So--

Instructor: That's really where network flow data really kind of comes to a shine, because it's predominately a retrospective data source.

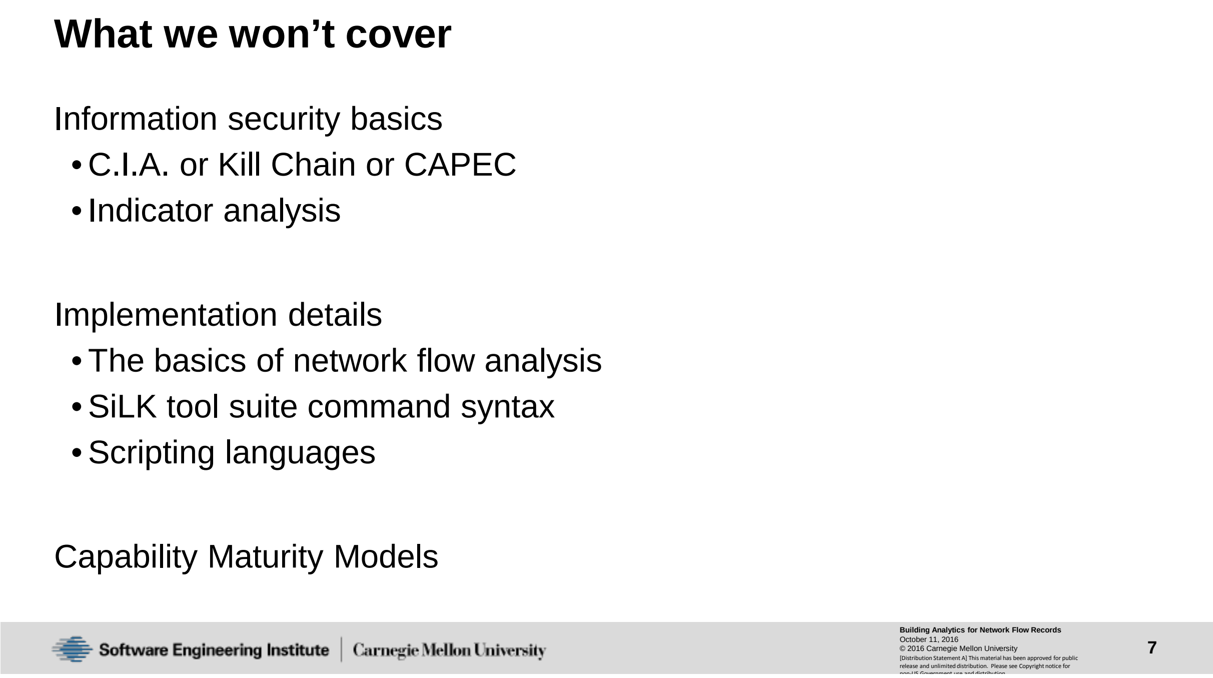
Instructor: Right.

Instructor: I mean, you can do analytics really close to real-time, but it's predominately retrospective. And that gives you a baseline of activity and also lets you observe the ongoing evolution of behavior in your network.

Instructor: Right. And I think if anyone has any experience with data warehousing or anything like that, it sort of falls into that area, right? We're looking at a repository of information over time, as Tim said. And then finally, the last thing is future. These are really things that, like, you know, are predictive analytics. Things to help you make decisions. They're really driven for making decisions, I think, is the idea behind the future analytics. But I think the point we're trying to make here is that there's a variety of ways that you can do analytics, but today we'll mostly be focusing on past.

Instructor: Yes, I think that's true.

What we won’t cover



What we won’t cover

Information security basics

- C.I.A. or Kill Chain or CAPEC
- Indicator analysis

Implementation details

- The basics of network flow analysis
- SiLK tool suite command syntax
- Scripting languages

Capability Maturity Models

Software Engineering Institute | Carnegie Mellon University

Building Analytics for Network Flow Records
October 11, 2016
© 2016 Carnegie Mellon University
[Distribution Statement A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

7

**007 Instructor: Right. Okay. What we're not going to cover are really some very broad-scale, you know, information security basics such as the basic models of threat.

Instructor: Right.

Instructor: The basic models of an attack. We're assuming that a lot of the people attending already have a particular interest they want to explore, particularly with respect to security. We're also not going to dive into implementation details.

Instructor: Right. Yeah. There's a lot that you can do with implementation. There's a variety of different technologies that we can make use of. Different, you know, types of programming language. And we're really just going to kind of focus on the process today.

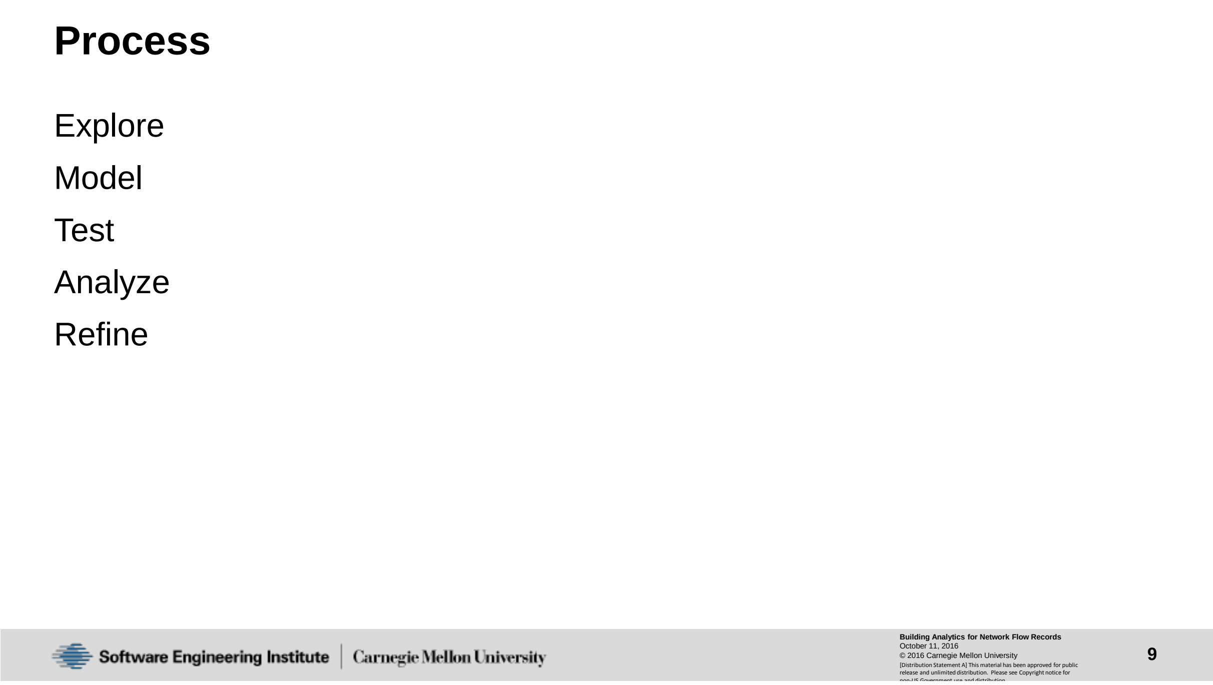
Instructor: Yeah. And we're also not--we've mentioned SiLK, but we're not going to dive too deep into the semantics of the SiLK suite. Examples will show it, but that's the case. And we're also not really looking--we've mentioned maturity--but we're not really looking to build a capability maturity model for analytics.

Instructor: Right. I mean, SiLK is pretty--or, you know, the SEI and CERT are actually known for CMMI and RMM and that kind of stuff, but we're not really going to be focusing on them at all in this presentation.

Moderator: Can we cover quickly what is SiLK exactly? You guys have mentioned that a couple, so for anybody new to the S--what exactly is SiLK?

Instructor: SiLK is a suite of tools designed to collect and analyze network flow data. Network flow data is aggregated packet header data. And SiLK's been around since 2003 or so. It is an open-source implementation of a SiLK analysis, or a flow analysis and collection suite. And there's about 30 to 50 tools in the suite that give you a lot of capabilities to both pull data off your network and make use of the data when it's there.

Process



Process

Explore

Model

Test

Analyze

Refine

Building Analytics for Network Flow Records
October 11, 2016

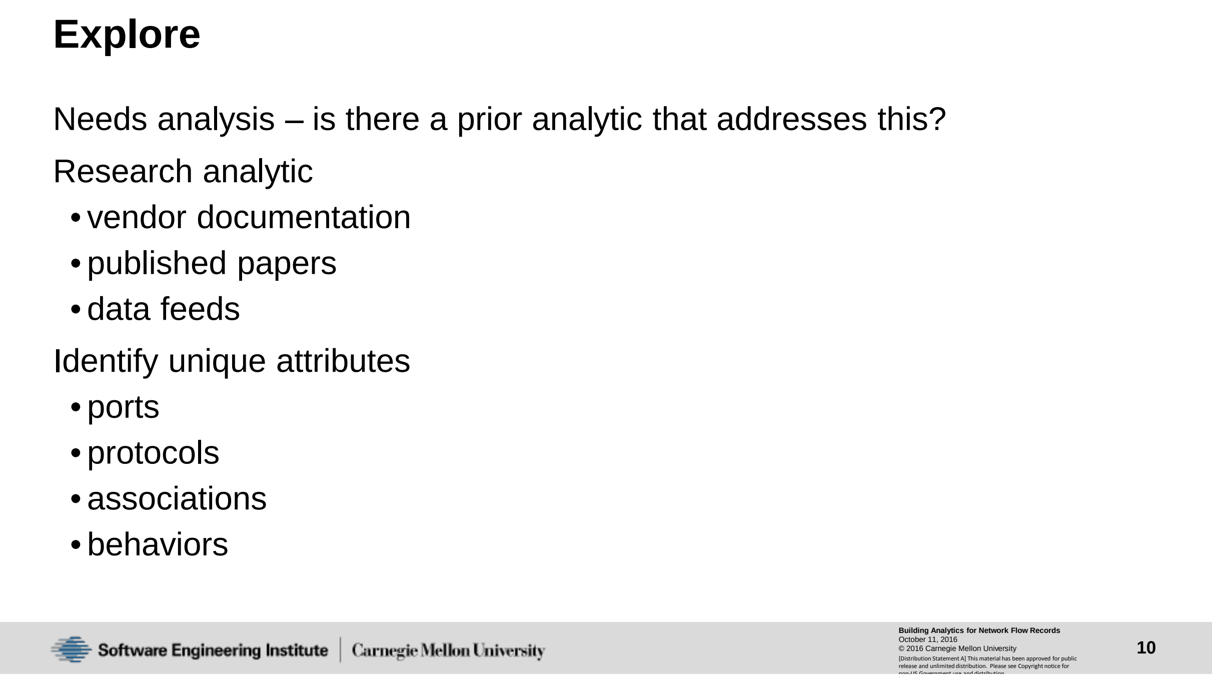
9

**009 We'll just start diving into what we think is the analytic, you know, process that we think best epitomizes the flow analysis process. So we've kind of laid it out here as a number of different steps, right?

Instructor: Yeah. Basically a five - step process. The first initial step is Explore. Then Model, then Test, then Analyze, then Refine. And we'll step through each of these one at a time.

Instructor: Right. Okay.

Explore



Explore

Needs analysis – is there a prior analytic that addresses this?

Research analytic

- vendor documentation
- published papers
- data feeds

Identify unique attributes

- ports
- protocols
- associations
- behaviors

Software Engineering Institute
Carnegie Mellon University

Building Analytics for Network Flow Records
October 11, 2016
© 2016 Carnegie Mellon University
[Distribution Statement A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

10

**010 So that brings us to Explore then. This is kind of the first step. What we're really trying to do here is basically explore your data, right? Where we're talking about finding out what exactly is in there, doing some needs analysis as well. So the idea here is that you're trying to find if you're already developed an analytic towards this or if there's a publicly available one, right? It's always good to not reinvent the wheel, right?

Instructor: Right. And there's a number of different places where there are published analytics available, including the FloCon proceedings that are available off the SEI site. But you're exploring not only the data, you're exploring analytic methods.

Instructor: Right.

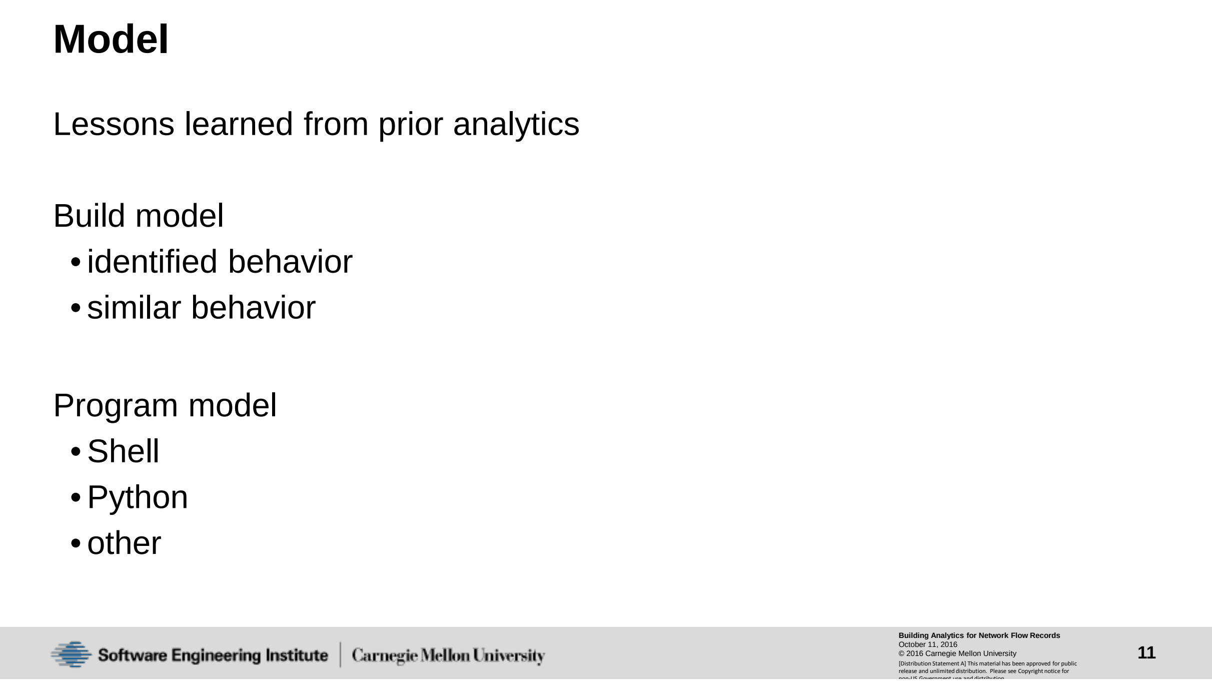
Instructor: So first of all, what's the particular problem that we're trying to solve? Then what are some candidate solutions we can use to try and both sharpen the problem and produce a better quality solution? And a lot of this deals with finding what aspects of the data really pertain to the problem we're going to solve.

Instructor: Right. I think truly understanding your problem set goes a long way into figuring out exactly what, you know, fields you might be interested in.

Instructor: Yeah. But they often break down to the things like ports, what--are there characteristic ports associated with the traffic we want to get? Are there protocols that we want to deal with? Are there particular relationships among hosts that can form an association that we're concerned about? Or are there broader-scale behaviors, patterns that you see in the flow between stimulus and response and further stimulus and response, that we could make use of?

Instructor: Right. I think one of the important things about what you just said is that last point, right. So flow really isn't often able to kind of point you to the smoking gun, right. With flow, we're kind of looking for where things might be, and these long-term large-scale statistical analyses can really help us, you know, pinpoint where we might want to look later.

Model



Model

Lessons learned from prior analytics

Build model

- identified behavior
- similar behavior

Program model

- Shell
- Python
- other

Software Engineering Institute | Carnegie Mellon University

Building Analytics for Network Flow Records
October 11, 2016
© 2016 Carnegie Mellon University
[Distribution Statement A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

11

**011 Instructor: So that, that brings us to the Model section of our process then. So basically what you're trying to do here is pull in some lessons that you've learned from previous analytics. This could be basically identifying additional sources for where you might get analytics from. This may be identifying things that just didn't work out last time when you created your analytic. Maybe you, you know, you were too targeted on Port 80 for web traffic when you really should be looking at maybe 80 and 4443 and 8080 and that kind of stuff, right Tim?

Instructor: Right. And frankly, technology's always changing. I know that sounds like a rubric, but it's basically, there's a constant evolution of behavior across the internet. New tools, new techniques are being used. And the attacks are also changing. So we need to be able to be, handle, some of that dynamism. And we'll also see this is sort of a landing point for something we'll talk about later, which is the Refine step.

Instructor: Mm-hm.

Instructor: Where you're not just changing the code, you're changing the basic model by which you do the analysis.

Instructor: Right. So maybe we've learned that an adversary likes to do things in a particular way and after we've created that analytic and used it, we come back during the Refine step with new information that we've learned from, you know, some new TTPs from that adversary and we change the model to reflect that.

Instructor: And then you build your model. You build your model out of what behavior you've identified and what similar behaviors exist.

Instructor: And then oftentimes you have a choice at this point, "Are we going to start with the very specific case and generalize up?" or, "Are we going to start with sort of a generalized case and exclude behaviors that we're not interested in?"

Instructor: So is that in reference to sort of a top-down or bottom-up approach?

Instructor: Correct.

Instructor: And then once we've got the model built and to the point where we can think it through, then we start to build, actually, program the model.

Instructor: Right.

Instructor: Because the goal is to have something which is readily repeatable and in computers that means program.

Instructor: And different organizations will choose different ways that they end up doing this, right, depending on the skillset of the people that they have. You know, they might just go straight for Shell programs. Maybe they have some people that have some pretty good Python knowledge so they go into that. In particular, SiLK allows us to do kind of either of those, right. We can focus on the Shell programs if we want, or if we wanted to we can dive into some Python code using something like NetSA Python or something like that.

Instructor: Well, there's also a lot of support for incorporating additional specific analyses right within the tool by using Python plug-ins.

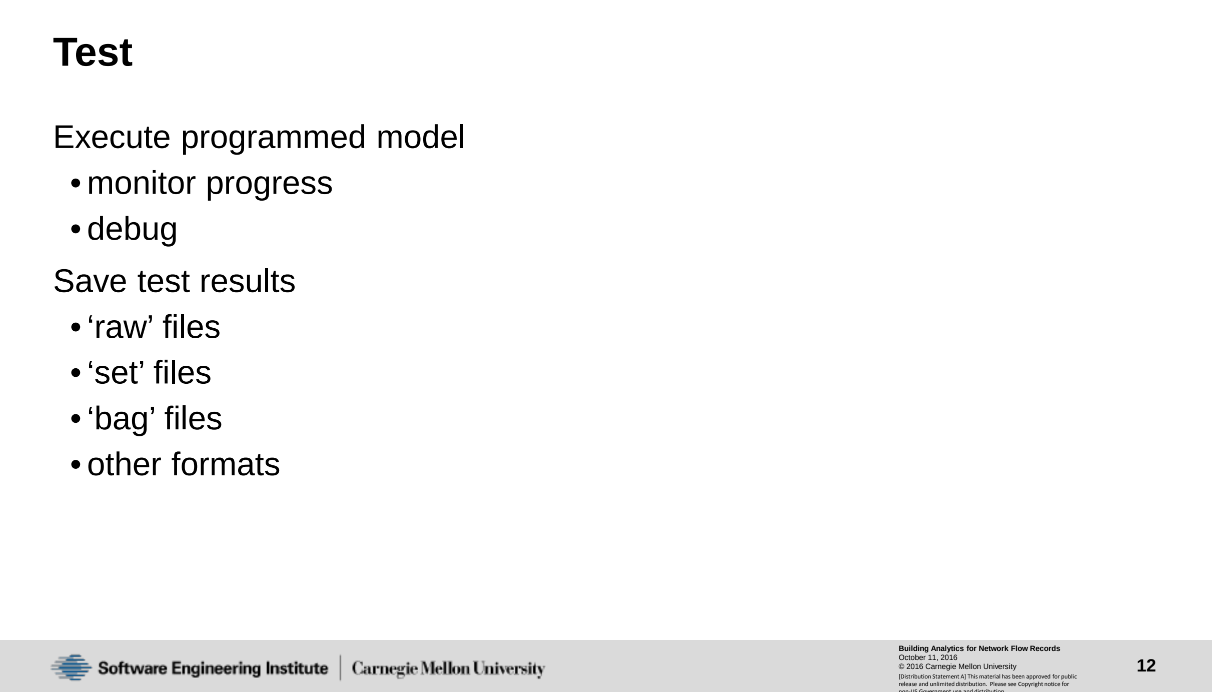
Instructor: Right. Exactly.

Instructor: And--or C plug-ins, of that matter. We could certainly use C, implement a lot of this, with C code. But that's often more overhead than organizations want. They want it to be more dynamic and more easily rolled out, so...

Instructor: Right.

Instructor: Shell or Python are really popular solutions.

Test



Test

Execute programmed model

- monitor progress
- debug

Save test results

- ‘raw’ files
- ‘set’ files
- ‘bag’ files
- other formats

Software Engineering Institute | Carnegie Mellon University

Building Analytics for Network Flow Records
October 11, 2016
© 2016 Carnegie Mellon University
[Distribution Statement A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

12

**012 Instructor: Mm-hm. Okay. And that brings us to the next step in the process that we've laid out here, which is the Test step, which is something that I feel is particularly important but tends to get glossed over a little bit. You really don't know how your analytic is going to perform unless you have a good set of test data, you're identifying maybe some kinds of edge cases. And really, I think, from a process standpoint, when you're talking about Test, it does kind of help to have another set of eyes. Right, Tim? I mean, there's certain things that you're just going to miss if it's your code.

Instructor: Right. And you want to be able to work within a team environment and have people review your code and bounce it from. But you're correct. We really want to define not just does it work on the data we have, but are there edge cases that could cause it not to work? Are there things that are similar that we're inadvertently alerting on? And this often gives us some feel for where the code is weak or strong, where the model may need to be tweaked and refined.

Instructor: Mm-hm. Yeah. And I think even just understanding maybe the particular use case you're targeting and how you might need to be aware of what sensors you should be focusing on, right. So if we're talking about some type of profiling or something like that where we're concerned with the edge of our network, we don't really want to include the sensors in that analytic from inside if we have that available, right. So even just understanding the limits of your analytics is a good part about the Test step.

Instructor: Yeah. The other thing that happens at this stage too is rather than really being heavy-duty production, we're more or less operating in a debug mode. Which means you save the intermediate files.

Instructor: Right.

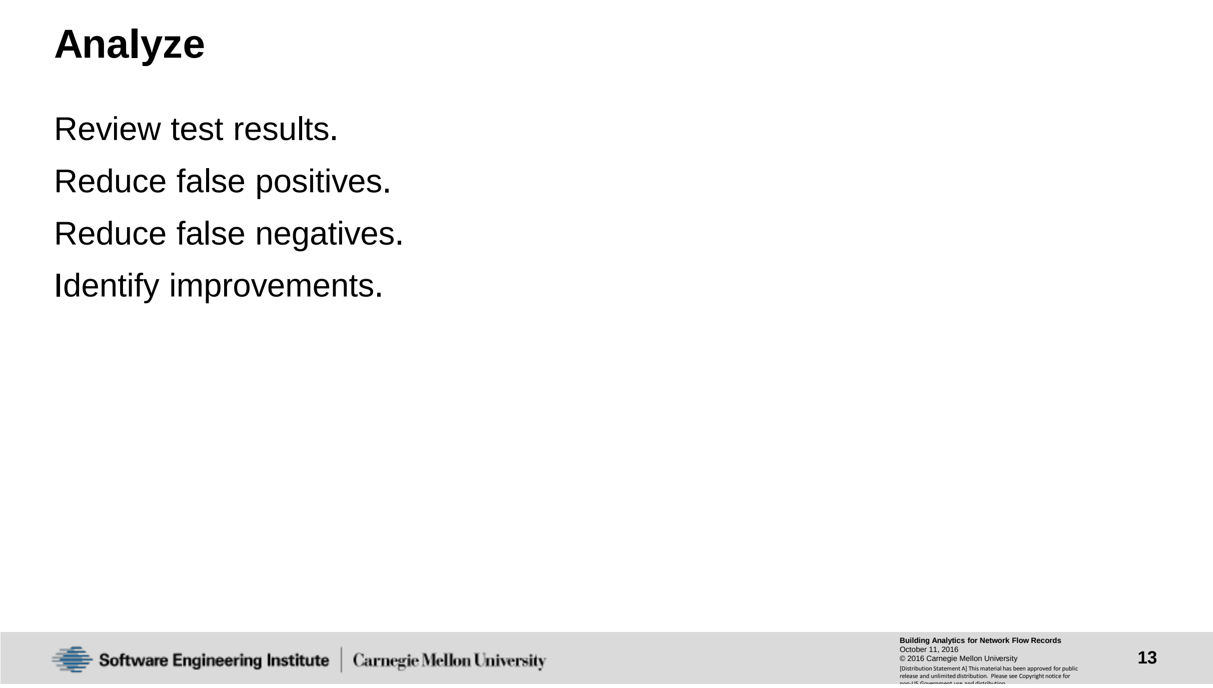
Instructor: And these intermediate files may include raw files, which in SiLK's case are binary stored data files. They may also include things that we extract from the raw files, such as sets of IP addresses or sets of IP addresses with associated counts. Those are called bags. And there may be some other formats. There may be CSV output that you want to display and peruse temporarily. There may want to be, you know, some routing data that gets incorporated.

Instructor: Well--

Instructor: But you want to be able to identify the traceability between these intermediate outputs so that you get a feel for where, if the model is breaking down, where the model is breaking down.

Instructor: Right. Okay.

Analyze



Analyze

Review test results.

Reduce false positives.

Reduce false negatives.

Identify improvements.

Software Engineering Institute | Carnegie Mellon University

Building Analytics for Network Flow Records
October 11, 2016

13

**013 So that brings us to Analyze then, after Test. So for this, we're really basically trying to review all of the results from the previous steps, right. So we're taking a look at our test results and making sure that things are kind of mapping to what we think they will be, but we're also trying to identify whether or not we are seeing false positives or false negatives, right?

Instructor: Right. Network flow data, as you've kind of referred to already, is very indicative. It's not very provable.

Instructor: Right.

Instructor: So you can't necessarily find a smoking gun. You can find things that raise suspicion. The advantage to this is it's very inclusive.

This advantage is some of the things that look like smoking guns aren't really guns at all or even smoking all that hard. I mean, you must've run into cases where you started to develop an analytic, applied it to a set of data, discovered some things and then realized, "That's not really what we're looking for."

Instructor: Right. Yeah. I mean, it comes up more often than you would think. And I think the whole iterative nature of this process really comes into play here, right. So we're identifying some of the failing points of our analytic and moreso than just that though. We're kind of sort of going back to the modeling stage, right, and the exploring stage here where we're trying to rework how we're thinking about the analytic in our head at this point and wondering why we're seeing some of the results that we are.

Instructor: Or, alternatively, why we're missing some of the things that we are.

Instructor: Right.

Instructor: If we're dealing with known data and throwing it against the analytic, we really want to be on the alert for what you're missing as well as what you're seeing. So this typically--alerting on something that you shouldn't alert is referred to as a false positive. Failing to alert on something that you should see is a false negative.

Instructor: Mm-hm. Right.

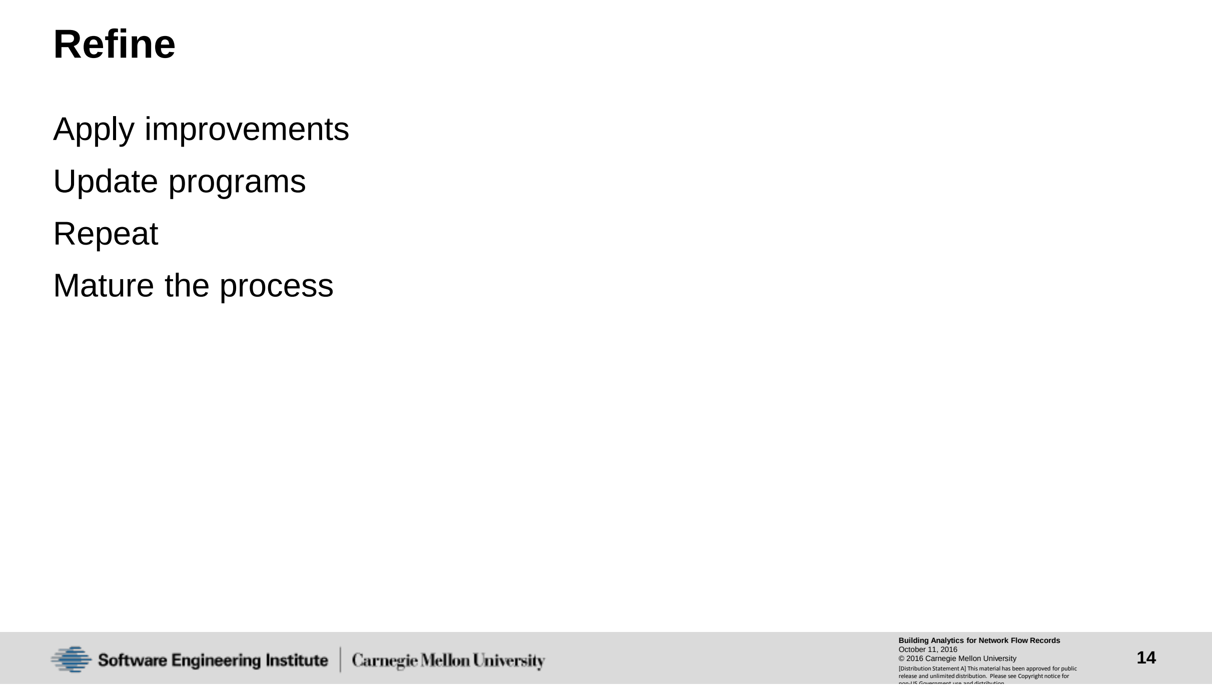
Instructor: And both are pretty serious. Because your false positives, that's going to overwhelm your operation staff. They're going to spend their time chasing down and eliminating things which are, in fact, benign, where false negatives, they're going to fail to get notification of things that could be malicious.

Instructor: Yeah. I think though that, you know, many organizations are aware of sort of the false positive, false negative thing. And one of the things that a lot of organizations do is they sort of create a threshold for themselves of what's acceptable, right. So this is just kind of something that you have to learn over time as you mature your process, as you mature your analytics. Like, we're only willing to accept, like, five percent of false positives, right, or, you know, a certain number of false negatives. And it's just the ongoing improvement that goes into this model here.

Instructor: And then you're looking at improvements both--you're right--both at the code level and at the module, model, level. And really trying to think through both how you've implemented it and what it is that you've implemented.

Instructor: Right. Yeah.

Refine



Refine

Apply improvements

Update programs

Repeat

Mature the process

Software Engineering Institute | Carnegie Mellon University

Building Analytics for Network Flow Records
October 11, 2016
© 2016 Carnegie Mellon University
[Distribution Statement A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

14

**014 So that brings us to the Refine step then. So really, I think one of the great indicators of a mature process is one that is, you know, driven from higher up to constantly be refining, right. You want to make sure that things just aren't archived away and just run and then forgotten about. And that knowledge of what that analytic is doing is forgotten. You know, whether that's because the person that initially created it left or whatnot. Really what we want to kind of drive home here is that you get to the Refine step, you're constantly improving these things, right? You need to go back up to the steps. You need to make sure that that knowledge of that analytic doesn't just disappear. Right? And also that it gets better, right?

Instructor: Well, and again, as the organization evolves as well as the technology and the threat, you want to make sure that that decision, the decisions that this analytic is meant to support, are in fact still current decisions.

Instructor: Yeah.

Instructor: Once the decision point is past, sometimes the temptation is to keep running the analysis and display it, when really that's not the most pertinent way of looking at the data anymore.

Instructor: Right. Yeah. So it goes all the way back to the first step at that point where you're sort of analyzing the landscape and saying, "You know what? We thought that this was important at some point in time to our organization, and it turns out that we don't really need that to drive decision-making anymore."

Instructor: Or it was important, but we've dealt with it.

Instructor: Yeah.

Instructor: And now we're monitoring more, "Are our solutions effective?" rather than dealing with that.

Instructor: Yeah.

Instructor: And so you're looking for timelines. You're looking to trend the results and decide, you know, "What is going on?" If we're always very static in terms of the results, that's a bit worrisome. Because that implies not much is changing with respect to our network threat. When we know that the behaviors are very dynamic.

Instructor: Mm-hm.

Instructor: So are we missing something? Are we getting complacent? And saying, "Yeah, we got an analytic to detect that. That's going to alert us if things are wrong," not realizing that the intruders have changed their game.

Instructor: Right.

Instructor: Changed the way that they act in such a way that it's evading our analytic.

Instructor: Right. So yeah. Ideally you want to constantly bring new information into this table. So you're not just, you know, analyzing your analytics. You're analyzing the information sources that go into creating analytics, right? So let's say that you've built up a set of sites that you monitor for network flow vulnerabilities or something like that and you realize maybe some of those has fallen down in, you know, quality, so you might want to drop them off your list. Like, it's a constantly improvement of not just the analytic development but also of the ways that you're going about even identifying analytics.

Instructor: Right. And there's also some process maturity that goes on with this. If you do have good management buy-in and good management practices in place, management is constantly encouraging the staff to question their assumptions. To question, to seek out, better solutions. To become current, to stay current, with not just their knowledge of network security and with threat, but also with the ways that they are analyzing their network traffic. It's very easy to fall into what's referred to as the detection trap, which is we've got a detector, we get an initial burst of results, and over time that, those results, tail off. And the assumption is, "Well, that's no longer a threat because we're no longer seeing it."

Instructor: Right.

Instructor: When in fact, it's still a threat and it's still occurring. It's just it's shifted in a way that decreases the effectiveness of your protection.

Instructor: Mm-hm.

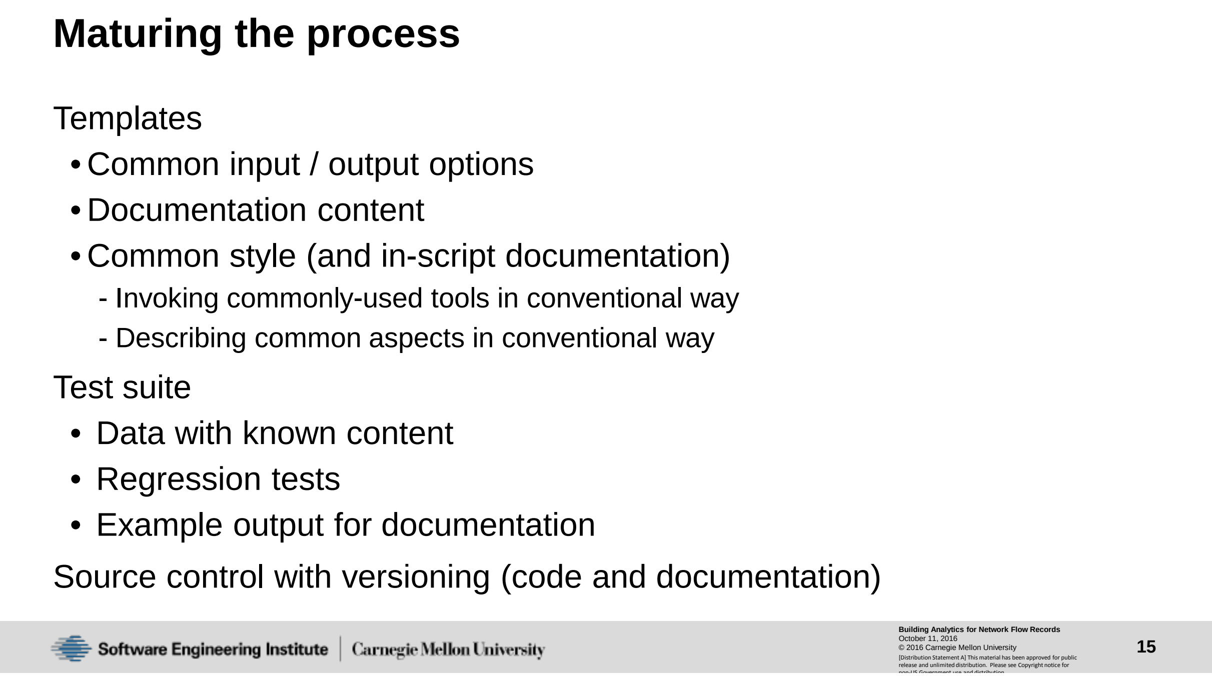
Instructor: And I think Tim kind of touched on this and I did a little bit as well, but the Refine step really does get the most value when it's coming from top-down, right? When management has bought into this improvement process, you know, there's only so much that an analyst can do. You know, many analysts are under constant pressure, there's a lot of work to do in this sphere, and if you don't have management kind of pushing for a continual improvement thing, it makes it very difficult to get to this step. Although we do, you know, we do think that this is a very valuable step, right?

Instructor: Oh, there's an awful lot of value to be extracted from it.

Instructor: Yeah.

Instructor: And like I said, we've mentioned. It is a feedback step. So you're feeding back to a lot of the previous steps. The point is to start to address the things in a way that the process is matured.

Maturing the process



Maturing the process

Templates

- Common input / output options
- Documentation content
- Common style (and in-script documentation)
  - Invoking commonly-used tools in conventional way
  - Describing common aspects in conventional way

Test suite

- Data with known content
- Regression tests
- Example output for documentation

Source control with versioning (code and documentation)

Software Engineering Institute | Carnegie Mellon University

Building Analytics for Network Flow Records
October 11, 2016

15

**015 And that also implies you're presenting things in ways that it's easy to reach the decision-makers or provide for a manageable process.

Instructor: Right. Yeah. I think coming up with some sort of common way of doing that is a great way of kind of getting to a more mature point, right? So we talk about things like templates a lot, right. You know, coming up with an idea of common inputs and outputs for your scripts, and things like--templates don't just apply to your scripts either, to your analytics, right. Templates also apply to other deliverables like documentation, right. Or maybe you're creating presentations based on analytic output, coming up with a template for that. It really helps to make it feel like it's a mature process, but it doesn't just feel like, it really is pushing you toward the maturity that we talk about.

Instructor: And it makes things clearer for the, both those that are producing the analytics and those that are consuming the results.

Instructor: Right. Yeah. I think it's very good for on-boarding. It's one thing that I've sort of found while doing this is that, you know, if we have a set of common templates for how you develop a script, that's great for somebody that doesn't have a lot of development experience, analytic development experience, right. There are certain things like how you handle the different inputs and outputs that some of the more senior members have kind of already dealt with, right. And if they put those into a common template that they share with some of the more junior analytic developers, that really helps bring everyone up, right? Where we're sort of trying to pull people towards the senior analytic developers, right?

Instructor: Another thing that really helps is if you can evolve a common set of test data.

Instructor: Yeah.

Instructor: And very frequently, just getting started and getting a good set of test data with some known content in there is a significant barrier. And you don't want to just reinvent that too often. You want to be able to continue to cycle on it and move forward. You also want to make sure that as you are changing the scripts you're not losing things that you wanted to keep. And that leads to support for regression tests.

Instructor: Right.

Instructor: And finally, you know, have good example output for the documentation so that as you're reviewing these tools and seeing which ones are, dealing with, you get a feel for, a very precise feel, for what it is the tool will give you as results.

Instructor: Right. And documentation is an interesting thing, because, you know, a lot of analytics are--in particular, SiLK analytics, the flow analytics that we're kind of focusing on today, you can have some pretty good in-line, in-analytic, documentation, right. So, you know, good comments, that kind of stuff. It's nothing really new, but the interesting thing is when you tie an analytic to an external piece of documentation as well, right. So you can explain a bit more in depth in that external documentation maybe in a way that you wouldn't in that script. And it gives you the ability to provide even some more detailed examples or how this analytic might be related to other analytics. And these are some of the documentation pieces that you might not want to put in comments in your scripts, in your analytics, right?

Instructor: Right. And also, since we're developing templates and since we're developing tools very iteratively, it really helps to have source code control.

Instructor: Yeah. And I think one of the big points I want to drive on this one is not just source control for your analytics. Like, your documentation needs source control as well.

Instructor: Yeah. You need to coordinate all of that, because the-- it's as your documentation changes, or as your analytics change, you want to make sure that you're retaining current documentation, and if there's a change you need to back out of or reapply in a different way, being able to recover a prior version both of the code and the documentation is very useful.

Instructor: Yeah. And the actually interesting thing about this too is as you get more mature with your analytics process and you use your source control more and more, you can actually track all through the commits and see different changes from other people and kind of learn things about the way that you've refined your analytics over time, right. Like, it provides a shared history that the organization might otherwise lose.

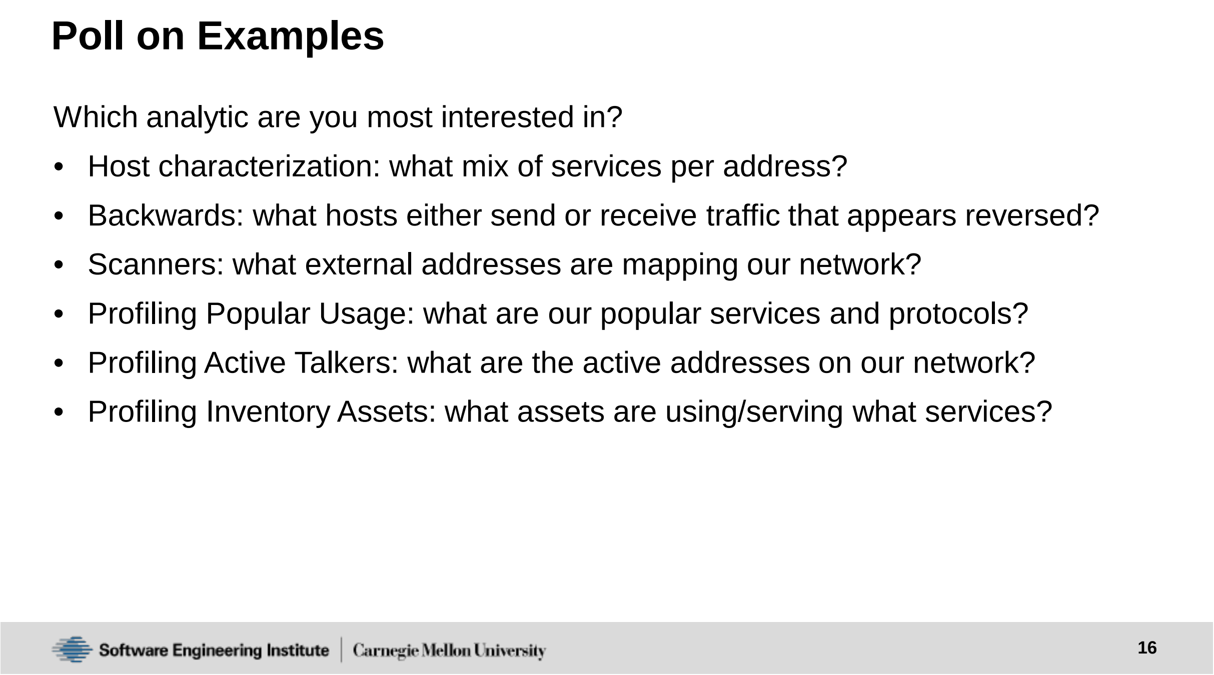
Instructor: Right. And particularly the comments associated with revisions.

Instructor: Right.

Instructor: And commits. Okay.

Instructor: Yep.

Poll on Examples



Poll on Examples

Which analytic are you most interested in?

- Host characterization: what mix of services per address?
- Backwards: what hosts either send or receive traffic that appears reversed?
- Scanners: what external addresses are mapping our network?
- Profiling Popular Usage: what are our popular services and protocols?
- Profiling Active Talkers: what are the active addresses on our network?
- Profiling Inventory Assets: what assets are using/serving what services?

Software Engineering Institute | Carnegie Mellon University 16

**016 Moderator: Okay. That's going to lead us to our third polling question, which is going to give us, drive the flow for example analytics for the rest of the presentation. So you'll see the question on your screen now. "Which analytic are you most interested in?" We got seven options there, so take about another 115, 20 seconds to vote. And while you do that, we're going to go to an audience question from John asking, "How do you verify that your model works? Do you try to generate a test data set with ground truth built in or some other approach?"

Instructor: Yeah. So it's a good question. I think that test data is always something that we sort of struggle with even here, right. And ground truth is something that's very difficult to establish in a lot of cases. But I think the idea is to try to put forward some sort of best effort when you're doing that, right. Particularly if you can make use of live data, you know. Just don't be testing it on a live production system. But if you can have some sort of mirrored data set or something like that that allows you to kind of do that, that would help a little bit. What other ideas do you have, Tim?

Instructor: Well, one, there is some data, particularly for if you're looking for a piece of malware and see a piece of malware running. Be able to run it in a sandbox and look at the structure which it offers. Look at the analysis that's already been done by some of the anti-malware vendors. Antivirus. And see if you can isolate those particular aspects that you would detect in flow-based analysis. Then actually generate data showing those analyses.

Instructor: Yeah.

Instructor: Then also showing things that are variants. And yeah, there's some data generation tools that are around that you can, you actually, produce flow data.

Instructor: Right. And we've had some varying success with making use of data generation tools. I think one of the big things is that people tend to see them as being not as realistic as it could be. But really what we're trying to strive with the test thing is you just want to make sure that ideally you're identifying what you think you're identifying, right. There aren't some things slipping by. And whether that's just taking a subset of your actual data or creating test data, I think that's best enough effort--

Instructor: Sure.

Instructor: --putting forward. And you can always iterate on that later. Right. If you identify something through the use of that analytic that you might've missed, it's a good thing to note that, "Hey, next time we come to make use of your test data we should make sure that it includes this."

Instructor: Well, in particular, if we can grab data from a real incident.

Instructor: Yeah.

Instructor: Where there is some, been some, investigation and there is some known truth.

Instructor: Mm-hm.

Instructor: That gives you a huge leg up and then using that as a springboard on which to incorporate other, fold other data into it.

Instructor: And I think that's actually a good point, because, you know, incidents, there's, you know, there's a lot of them. And we might actually have ongoing forensic analysis going right now in an organization on that. And what we can do is we can, you know, take the PCAP from that, you know, incident and convert it into flow and then test our analytics against that.

Instructor: Yeah. We've done that a couple times.

Instructor: Yeah.

Instructor: And it's interesting to do.

Instructor: Mm-hm.

Instructor: Finally, there are some data sets that are now public out there on the internet. Both sort of general data, the sort of day in the life of the internet type data. Or some very event-specific data, where it's like this is what a virus infection looks like, this is what a DDOS of a particular type looks like.

Instructor: Mm-hm.

Moderator: Okay. So let's get those results. We had 28 percent profiling inventory assets. At--

Instructor: Okay.

Moderator: --24 percent, we had host characterization, what makes those services per address. And then the next highest percentage was 17 percent, and that was profiling active talkers.

Instructor: Okay. So let's--

Instructor: Okay.

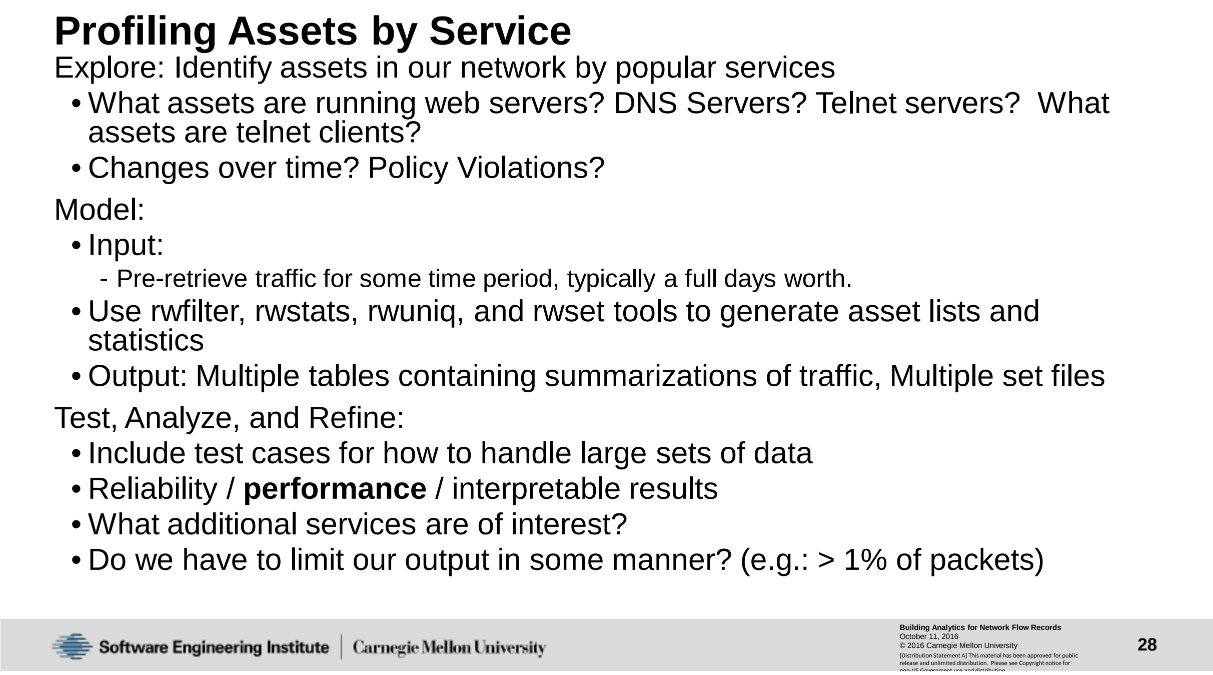
Instructor: Let's run through those kind of in that order. I'm not sure quite how far we'll get--

Instructor: So--

Instructor: --but we'll try.

Instructor: All right. So let's take a look at Profiling Assets by Service, I believe?

Profiling Assets by Service



Profiling Assets by Service

Explore: Identify assets in our network by popular services

- What assets are running web servers? DNS Servers? Telnet servers? What assets are telnet clients?
- Changes over time? Policy Violations?

Model:

- Input:
  - Pre-retrieve traffic for some time period, typically a full days worth.
- Use rwfilter, rwstats, rwuniq, and rwset tools to generate asset lists and statistics
- Output: Multiple tables containing summarizations of traffic, Multiple set files

Test, Analyze, and Refine:

- Include test cases for how to handle large sets of data
- Reliability / **performance** / interpretable results
- What additional services are of interest?
- Do we have to limit our output in some manner? (e.g.: > 1% of packets)

Software Engineering Institute | Carnegie Mellon University

Building Analytics for Network Flow Records
October 11, 2016
© 2016 Carnegie Mellon University
[Distribution Statement A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

28

**028 Moderator: Profiling Inventory Assets.

Instructor: Profiling Inventory Assets. Which is--

Moderator: Yeah. By you--

Instructor: --Spiceline, yes.

Moderator: By what assets are--

Instructor: So one of the issues that we actually have with this particular one is that the code for this is pretty long. So in your talk at the beginning, there are some text files in the--

Moderator: There's a Download Materials widget, yeah.

Instructor: Right. So in the Download Materials widget, there's a text file that is this code, right? And I believe it is the assets by service.txt. So if you can go ahead and open that up, you can follow along with basically what the code is. But we'll go over sort of how our model fits with that, our process fits with that first, before we kind of dive into the code a little bit.

Instructor: I will comment that although it's a .txt file, this is really a Shell script.

Instructor: Mm-hm.

Instructor: So you could, if you're running UNIX or LINUX, you actually can apply it. It's a bash script. For doing this activity.

Instructor: Yep. So you want to go ahead and start walking through the process, and I'll pull up the code over here so we can take a look.

Instructor: Okay. So when we start to explore the--explore it, what's the problem that we're trying to solve here? And that is we want to identify where the servers are in our network for popular services. So we want to be able to say, "Okay. Which assets on our network are providing web service? Which are providing DNS service? Which are providing Telnet, or acting as Telnet clients? And we're looking for basically not just the initial snapshot. We're looking for it in a way that'll allow us to do changes, watch changes over time, watch perhaps for policy violations, where service is being offered by a host that isn't authorized to do it. That's going on in there.

Instructor: Yeah. And ideally, the way that we kind of model this is we take a look at inputs, the small, high-level implementation detail, which is just kind of pointing to what tools we think we might need to find this, and then outputs, right. So inputs, the tools that we're going to use to discover this activity, and then the outputs that we'd like to get.

Instructor: Right.

Instructor: Right. So in this case, we're talking about pre-retrieving some traffic for a particular time period, and typically when we're doing profiling we like to kind of focus on a full day's worth of traffic, at least. Additionally, there's a num--

Instructor: I would jump in here. I mean, if you're trying to do a full day's worth of PCAP, that can be pretty extensive. You need to have a big disk available, particularly if your network is any size. Flow is much more compact. I mean, a single flow record is only a few bytes. And then you can--particularly when it's compressed. And then you--so you can hold a day's worth of data and you can process a day's worth of data pretty efficiently.

Instructor: Yeah. And I actually did run this particular script earlier today, and it completed fairly quickly, so just...

Instructor: About how big a network did you look at?

Instructor: Our network, actually, the SEI network. So not huge, but--

Instructor: Probably a thousand or so hosts.

Instructor: Yeah. So not small but not huge.

Instructor: Okay.

Instructor: But yeah. So then the way we kind of lay this out is we're taking a look at what tools we think we'll need to use for this. So in particular, we're looking at rwfilter, you know, to give us some way of paring down what we're interested in, rwstats and rwuniq, to give us some statistical or, you know, some, you know, top 10s, bottom ends, you know, kind of output. And we're also making rwset, right.

Which allows us to create, you know, IP sets, which are just a big list of IP addresses. And also it allows us to do some basic statistics on that and do, you know, set math as well.

Instructor: Right.

Instructor: So we think that these are the tools that we would end up using to create this analytic, and, you know, we could be off the mark, but in this case--

Instructor: Well, and it's useful to have not just text output saying, "Okay. Here's the servers that were identified with a particular service," but also to have these set files, because then in further analysis you can actually use the set files and say, "Okay. Of the web servers, how many were also doing this and how many were also doing that?"

Instructor: Right. And we can use it, since we're talking about profiling as well, we can kind of do set math on a day-to-day basis of discovering, "Hey, you know, which IP addresses that we're offering these services have dropped off? How many new ones are there available now?" Right. We can do that pretty easily if we keep the sets around.

Instructor: Right.

Instructor: Right. So that brings us to Output.

Instructor: And then once we have the output, okay, for test cases, we want to include large enough test data to where we get a feel for, you know, "Are there confusion factors being introduced?" So you generally do want to drive to go from live capture data if you can. Performance is often very much an issue with respect to these scripts, because you're running through the data several times.

Instructor: Yeah. In general, profiling is typically starting from a point where you're just looking at every IP address that pops up for a particular day, and then pairing it down from that. So that first pool is always one that you kind of want to worry about a little bit of how long is that going to take? And then after that we're doing some statistics on top of that large set of data. So performance is something to keep in mind with this.

Instructor: Next would probably become reliability. Are we correctly modeling the particular services that are involved in a way that let us recognize--well, particularly DNS. Is this a client-to-server communication? Is this a server-to- server or server-to-client? And there's slight changes in behavior with each one that it'd let us lay it out, but there are times when that changes. And in particular, several years ago was the Kaminsky bug. Prior to that point there was a lot of cases, lot of very characteristic port behavior associated with servers. That kind of changed at that point because we realized we needed less predictable ports. And so we needed to shift things in place. So you want to make sure that you're including ways that are reliable in the modern world.

Instructor: Mm-hm.

Instructor: With respect to e-mail. Some of the e-mail is going to occur over the normal ports. Some are going to occur off port by organizations that vary their e-mail traffic the way that e-mail traffic is implemented.

Instructor: Right. And these all kind of drive decisions of, like, how you're going to ultimately implement this, right? I mean, if you're making use of something like YAF, which is our collection application for that. The SiLK tools just natively understand. YAF has the ability to do some application determination as well. If you're not making use of that, you know, you're going to have to shy away from using that particular field. It's just something that goes into understanding your infrastructure and then understanding how that tailors what kind of analysis you're going to be able to do.

Instructor: Right.

Instructor: Right.

Instructor: And then at the point, you know, we implement it for a few services and then we realize, "Oh, we need more services involved." So the question is, "What traditional ones do we want to add?"

Instructor: Right.

Instructor: And finally, do we want to limit our output in some manner? So are we concerned about every possible service? Are we concerned with services that only constitute a, only exceed, some threshold for our traffic?

Instructor: Right.

Instructor: That really deal-- manage it.

Instructor: So with things like web, for example, I mean, you could always try to pull up web clients. But there's going to be a lot of web clients. So you might want to limit the number of output, you know. Maybe we're only interested in profiling web clients that do a particular set of traffic, right. Like, maybe they're more than one percent of packets, you know, web packets or something like that.

Instructor: So for those that do have the script open--

Instructor: Right.

Instructor: --are there some points that really particularly might be pertinent to this? I mean, we initially start off with basically slicing and dicing the data.

Instructor: Mm-hm. And if you take a look, if you are actually looking at the script, this is actually directly out of one of our other publications, the "Network Profiling with Flow," that our Austin Whisnant actually wrote. And in particular, we're taking a look at the Section 5 script from that document. But you should have it available to you through the webinar, so you could always open that up right now.

One thing to take a note of in this is that the first rwfilter call, in particular all the rwfilter calls, are really pulling from a file called sample.rw. This is a file that was created actually in an earlier analytic. So you can kind of see how one analytic will flow into another. In particular, this is the, one of the other profiling analytics, actually. I believe it's the profiling top five services analytic actually creates this file. Ultimately, what that file is is it's a pool of data for a day of every protocol. So protocol zero and up. And now that we have that, we've saved that around from an earlier analytic. And we don't have to reach back out to the repository to do a pool again now. We also have saved some things like, you know, sets from other analytics and we could make use of them in here.

So what we're, what I basically can do now, is we can kind of walk through a little bit from a high level of what's going on in this analytic.

It's pretty long. It's probably the longest one, so it wasn't exactly the best one to start on. But it is probably, it's obviously, the most interesting one for the audience. So basically from the get-go what we're doing is from our preset, our pre-pooled version of data, the sample.rw file, we start paring down web servers. We do that by taking a look at outbound web traffic with a source port. That's a web port. In this case, 80, 443 and 8080. And we're only interested in TCP, so, and, you know, our protocol is six, and we're, in particular, we're only looking for flows that have four or more packets, right. So we want to identify flows that have gone past the TCP handshake, right. We want to go so that they've gone and they've started exchanging data. It's a pretty general way of identifying a web server. You know, we've--it's a outbound traffic on, you know, the normal ports, and it's gotten past the TCP handshake, so it accepted it. It kept going, right.

So from that, then we can go ahead and we can limit it so that we're only interested in, through rwstats, we can pull out the source IP. So that gives us our list of now web servers. And we're limiting that by percentage. In this case, we want one percent. So otherwise we might get more than we want. This is what we were kind of talking about in the previous slide of sort of tweaking, limiting your output in some manner. Yeah.

Instructor: Right.

Instructor: And then from there we're going to go ahead and put that into a set, right. And we're going to do that very similar for a lot of these things, right. So if we drop down to web clients, you can see in this case we're now doing destination port instead of source port. So that way we're looking at the client versus the server. And again, we're going to, you know, put that into a set, right. And we can make use of these sets for output and for doing set math. So if you take a look at rwset tool line there, under the Web Client section, you can see that what we're doing is we've built a couple sets. We've built a TCP client set and a UDP client set and what we want to do is union them to get a big set of all web clients, right. So ultimately, that's sort of how we're going to walk through profiling each one of these services.

We also have some outlier identification here. So we have potential web clients, potential web servers. The idea is that we may be missing something in our initial rwfilter pool. So in this case, we're going to not limit that as much and say, "These might be web clients. There might be web servers," but we're not as confident as we were on the initial pools and the initial stats, right. We think they are. They could be of interest to people. To us, in particular. But we don't want to include them in our original output, right. Because we're not a hundred percent sure.

Instructor: Well, one of the difficulties you get a lot with flow analytics is it's easy to find something.

Instructor: Yeah.

Instructor: What's tricky is finding just the thing that you want with no extras.

Instructor: Right. So that's a good example of web clients and web servers. There's also e-mail here. And we do similar stuff with e-mail and DNS where ultimately what we're doing is we're pulling out based on ports and protocols and packets so that we, you know, identify things that have actually gone past the handshake. In particular, if we're looking at TCP. And we also, you know, put a lot of this stuff into different sets so that we keep them around. So like we said, the next time we run this if we run this today, we have those sets. We run it tomorrow, we can do an intersection between the sets. Maybe we write a separate analytic for that. So we don't have to run it manually. So we run this one, run a set analytic that does the set math, and now we have the differences between the sets of the two days.

Instructor: And it may also be useful not just to profile two days but to spot whether they're cyclical behavior.

Instructor: Right.

Instructor: Sometimes you see weekly patterns, sometimes monthly.

Instructor: Right. And one thing that we didn't really point out here is you could always take this data that you get from here and throw it into something like Excel or something like that. Or even use something like Rayon, which is another set of tools that we provide here for doing graphing or, you know, graphics basically. And you can take this output and format it in any way that you would want to present it to somebody else or even to yourself.

Instructor: Right.

Instructor: Right. So, you know, if you keep moving down, you can see that we've done some interesting stuff with DNS. We've identified that under the DNS section. We have a set of DNS servers, a set of DNS clients, and then we do an intersection between the set of DNS clients and the set of DNS servers and we get what we believe to be recursive DNS servers or iterative DNS servers. And that's just based on our knowledge of that protocol, right.

Instructor: Well, and also where the threat frequently lies.

Instructor: Right.

Instructor: So for example, one of the issues is are there open resolvers on our network?

Instructor: Mm-hm.

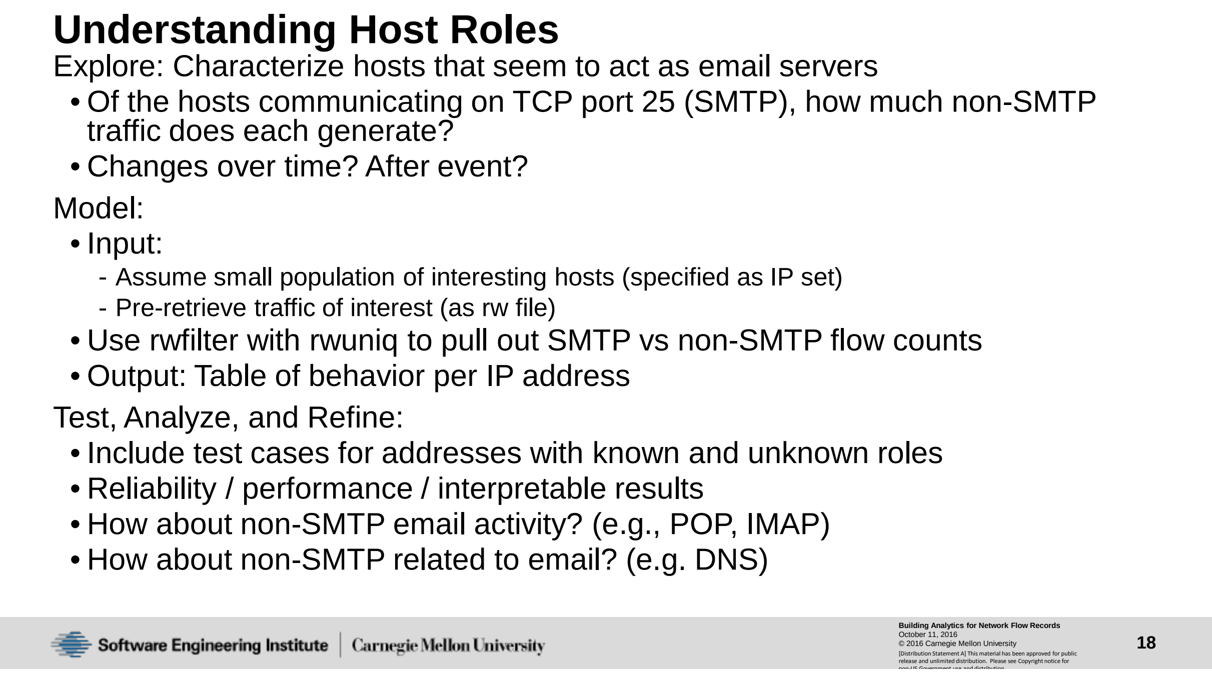
Instructor: Where's the traffic actually coming from and going to?

Instructor: Right. And one thing to note too with the profiling stuff is we're not really, we're not really looking to pinpoint just an individual threat really. What we're kind of doing with the profiling scripts is helping to get a baseline situational awareness of what's going on on our network. And so one of the ways to do that is to take a look at the services that we're running, right?

Instructor: Okay. So we've kind of discussed this one in pretty good depth. I think the second one--

Moderator: So we're going to go to the next one, which was--while you pull that up, it's 25 percent host characterization. And while they're pulling that up, I want to make sure everyone knows that they're invited to FloCon 2017. FloCon is the 13th annual open forum for large-scale network analytics.

Understanding Host Roles



Understanding Host Roles

Explore: Characterize hosts that seem to act as email servers

- Of the hosts communicating on TCP port 25 (SMTP), how much non-SMTP traffic does each generate?
- Changes over time? After event?

Model:

- Input:
  - Assume small population of interesting hosts (specified as IP set)
  - Pre-retrieve traffic of interest (as rw file)
- Use rwfilter with rwuniq to pull out SMTP vs non-SMTP flow counts
- Output: Table of behavior per IP address

Test, Analyze, and Refine:

- Include test cases for addresses with known and unknown roles
- Reliability / performance / interpretable results
- How about non-SMTP email activity? (e.g., POP, IMAP)
- How about non-SMTP related to email? (e.g. DNS)

**018 And this year's conference is going to take place January 9th through 12th, in San Diego, California. And basically FloCon's a four-day conference geared towards operators, analysts, researchers, and anyone else who wants to apply next-generation techniques, analyze large volumes of network data. And by attending today's webinar or watching the archive, you get 10 percent off by registering with the code of FLO-WEB17. So when you're on the registration page for FloCon, add in that code at the registration site and you'll get a 10 percent off of that. And great conference. Both Matt and Tim are speakers and instructors at the conference, so--

Instructor: Right. There is a training day. The first day is actually a training day.

Moderator: Yeah.

Instructor: Which is often very rewarding both for the presenters and for the attendees.

Instructor: I just wanted to note that Tim has actually been at every single FloCon now?

Instructor: Yeah. I'm--I will be 13 for 13, yeah.

Moderator: Thirteen for thirteen.

Instructor: So--

Moderator: Which is impressive.

Instructor: I unfortunately do not have that streak going. I've missed a few years. But hopefully I can maybe match that eventually.

Instructor: I figure when I get to 50 out of 50, they'll retire the...

Instructor: That's right.

Instructor: Okay.

Moderator: So we're on the host characterization.

Instructor: All right.

Instructor: Okay.

Instructor: And let's go a little bit more briefly here. We may be taking an initial characterization of hosts like the one that we produced in the script we just discussed, and now we want to refine that characterization. So particularly for e-mail. Okay. How much e-mail traffic is a host doing versus non-e-mail traffic?

Instructor: Right. Right. So for example, let's say that we have a specific e-mail server set up. And we just want to make sure that it's not doing weird stuff. Maybe it's communicating on VPN, you know, or doing something like that. And we're trying to identify a little bit more about that host.

Instructor: Well, partic--also if it looks like a client that's providing e-mail service, that's somewhat suspicious.

Instructor: Mm-hm.

Instructor: And worth at least looking forward, looking further at. In addition, there are times when it may blur behaviors. So if we get a host which is acting as an e-mail server, a web server, a DNS server, and things like that, is that really one host or is it a gateway.

Instructor: Right.

Instructor: Or a NAT host. That we're somehow false flagging. And so we need to deal with cases like that. And again, you want to know whether things change after time. If we do have an event on our network, an intrusion, has somebody put in place code that really isn't authorized? And we'd like to be able to use network traffic to try and identify that case to make sure we've cleaned up after the event properly or are dealing with a ramification. And there the model is, "Okay. We've got a small initial population of interesting hosts maybe coming out of the profiling by--"

Instructor: Service. Yeah.

Instructor: Service. And as well as a traffic capture that we've grabbed into a file. And then we use, we're going to filter that traffic, and then we're going to re-characterize it based on what services we observe from it. So we're going to use the wrfilter tool to isolate out a particular host and we're going to use wruniq to profile all the services that are involved in that host.

Instructor: And in particular with this one, the example that we've provided where we're taking a look at SMTP versus non-SMTP flows. Right.

Instructor: Yeah. Exactly. And then the table is simply lists of behaviors per IP address, presented in a one line per IP address fashion so the numbers are easily comparable.

Instructor: Mm-hm.

Instructor: And then once you've got that, again, you'll want to go through a test process. This one's probably not as much a performance issue because we've already narrowed down the scope enough to be able to run very, very quickly to our saved traffic. Here the result may be reliability or interpretability.

Instructor: Yeah. So one thing to keep in mind is that as you're iterating over these scripts, getting input into what kind of output you're producing from other people will really help you sort of get it to a better spot that it's more general use, right. You don't want it to just be so pinpointed in its output that really only one person or a couple people can use it. Ideally you can output it in some sort of CSV format or something like that that can be used in other analytics more easily.

Instructor: And brought into tools like Excel and--

Instructor: Right.

Instructor: Or plotting tools or what have you.

Instructor: Mm-hm.

Instructor: Also there's the case of, you know, what about other non-SMTP e-mail protocols?

Instructor: Right. And this is kind of pointing towards the Refine step. So if you take a look at our initial analytic for this versus sort of what we're calling out as iterative things that you could now do, right. So ideally you'd iterate on our script and add things like looking for POP or IMAP or something like that.

Instructor: Right. Or the other aspect is is the other service that we see from our mail servers expected?

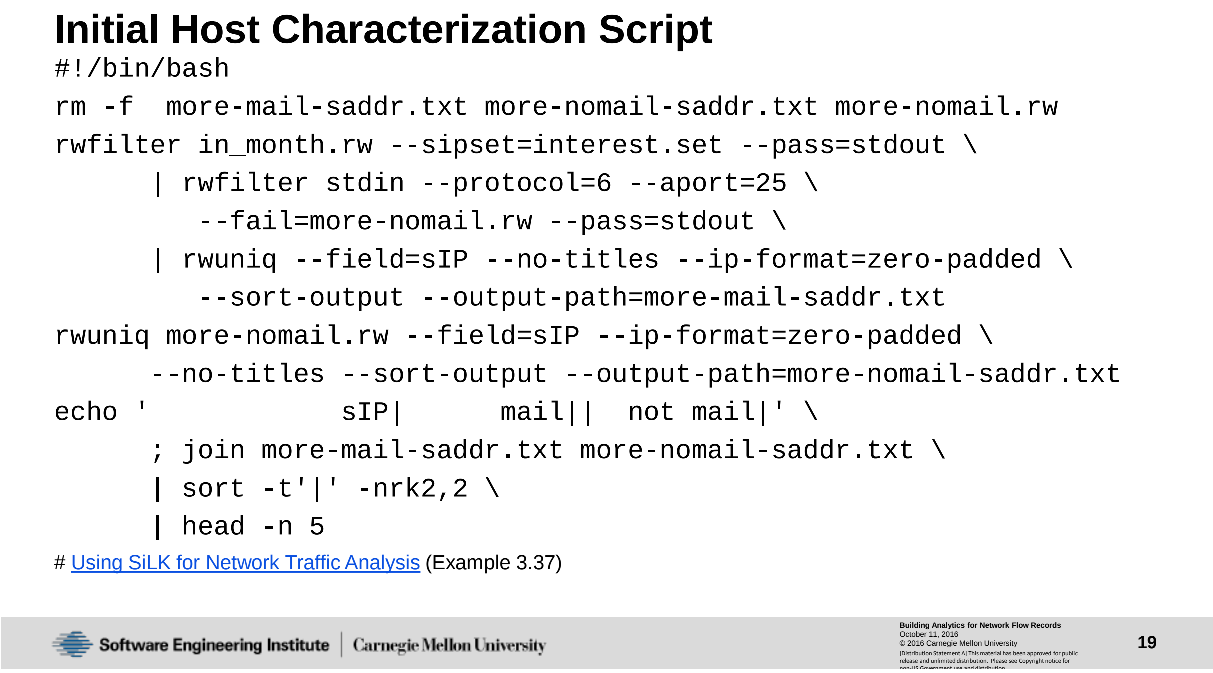
Instructor: Mm, yeah.

Instructor: So frequently there may be DNS traffic associated with an e-mail server where it's checking the validity of data, the validity of domains. There may be some administrative traffic, which is expected. And in log file output activity to a syslog server or something like that. We might want to characterize the expected versus unexpected. And again, that's a further refinement to help us deal with an understanding of what's really going on with these hosts.

Instructor: Right. Yep.

Instructor: And so now let's take a look at the code.

Initial Host Characterization Script



Initial Host Characterization Script

```
#!/bin/bash
rm -f  more-mail-saddr.txt more-nomail-saddr.txt more-nomail.rw
rwfilter in_month.rw --sipset=interest.set --pass=stdout \
      | rwfilter stdin --protocol=6 --aport=25 \
         --fail=more-nomail.rw --pass=stdout \
      | rwuniq --field=sIP --no-titles --ip-format=zero-padded \
         --sort-output --output-path=more-mail-saddr.txt
rwuniq more-nomail.rw --field=sIP --ip-format=zero-padded \
      --no-titles --sort-output --output-path=more-nomail-saddr.txt
echo '            sIP|      mail||  not mail|' \
      ; join more-mail-saddr.txt more-nomail-saddr.txt \
      | sort -t'|' -nrk2,2 \
      | head -n 5
```

# Using SiLK for Network Traffic Analysis (Example 3.37)

Software Engineering Institute | Carnegie Mellon University

Building Analytics for Network Flow Records
October 11, 2016
© 2016 Carnegie Mellon University
[Distribution Statement A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

19

**019 This one will actually fit on a slide. And by the way, this actually came from the SiLK Analysts handbook, which is formally titled, "Using SiLK for Network Traffic Analysis."

Instructor: And the link is available right in the presentation if you'd like. Otherwise it's on the Tools site, so--

Instructor: Yeah. And the Tools site is available. It's there. This one actually starts off by clearing off some stuff. And this is often something we do with analytics. To make them more robust we want to make sure that we clear off any prior results which are present.

Instructor: Right. And this is sort of something that you as an organization kind of need to define and deal with, right. So what we've done here is a very simple way of handling that, right? But it's possible that we don't want to delete or overwrite anything else that was there, which is going to kind of drive how you do your cleanup operations differently as an organization, right. So this is a simple example, so we're just going to delete it. But you might want to keep it around.

Instructor: Or you may want to use a utility like mktemp or mkstemp to produce unique filenames for every run. And that way it's less likely that you're going to see corruption of data or overwriting data and things like that. Then we use the filtering to pull out e-mail and non-e-mail data. And here's a case where we're actually filtering twice to get things in the location we want. Then we summarize using rwuniq. And--

Instructor: Yeah. And you can kind of see. So Tim pointed out, we're filtering twice. We're actually failing some stuff here with an rwfilter tool, because we want to keep that around.

Instructor: Yeah.

Instructor: We're not getting rid of it.

Instructor: And finally, we've produced some output. And we use a Join command to bring together the correlated results and produce a nice columnar output.

Instructor: Yep.

Instructor: That's in place. And a little bit of sorting and a little bit of summarization.

Instructor: Yeah. And ultimately what you end up--

Instructor: Not really sure we want to keep the head-n5 on there, but, you know.

Instructor: Yeah, well, if we were going to show output from this and we needed it to fit on a slide, that's what you would do.

Instructor: Yeah. Again, the concern is do you want to track your top five or do you want to move things forward?

Instructor: Right.

Instructor: Yeah. Alrighty. We've only got about three minutes left. So--

Moderator: Yeah. So--

Instructor: --rather than dive into another--

Moderator: Yeah. I want to say, rather than dive in another one, the queue is open for any questions. Otherwise we'll start wrapping things up here. Because, like you said, we can't get into another script. But just a reminder to everybody that's--the archive of this will be available by tomorrow. You can login to watch with the same registration or the same e-mail that you logged in today. By tomorrow morning the archive will be up and available. Again, the scripts that they're referring to are actively available in the Download Materials tab. So you can catch everything there.

Instructor: And actually, there have been several that we didn't actually get a chance to explore, but it's discussed in some slides, which will be available in the PDF. Exactly.

Instructor: And additionally, me and Tim have both mentioned this, but the slides will refer to where those scripts publicly came from. So if you wanted to get even more depth you can go into the handbook and read about the scripts in more depth. Or you can check out the Network Profiling document as well.

Moderator: Excellent. So we're going to wrap it up there, folks. Just a reminder, our next webinar will be October 19th, and the topic will be "Security Practitioner Perspective on DevOps for Building Secure Solutions" by Hasan Yasar and Lackey.

And you'll get an invite to attend that. Again, thanks everyone for their time today. We appreciate for the presentation by Tim and Matt. Great job.

Instructor: Thanks for having us.

Moderator: And we'll hope to see you soon in a webinar again in the near future.

Have a great day.
Thank you.

SEI WEBINAR SERIEW | Keeping you informed of the latest solutions

SEI WEBINAR SERIES | Keeping you informed of the latest solutions

Software Engineering Institute | Carnegie Mellon University