2019 | 19

Product Security Incident Response Team (PSIRT) Maturity Document

Table of Contents

PSIRT Maturity Levels to demonstrate Operational Capability and Maturity

Maturity Level 1 (Basic) - The Beginning is a very fine place to start

Preface:

There has long been an interest in cyber incident response; events in recent years have only increased that interest. In 2013 FIRST initiated efforts on creating Services Frameworks with a focus on CSIRT operations. Following the publication of the CSIRT Services Framework, the Product Security Incident Response community met under the FIRST umbrella to craft a PSIRT-centric Services Framework that would address the unique challenges that those teams experience. While CSIRTs and PSIRTs share common behaviors and activities, the former is focused on protecting an organization's infrastructure while the latter responds to threats and flaws in the organization‘s products. The Board appreciates the efforts of the product security community and want to thank them for defining their Services to educate all.

This document presents a series of use cases as well as a high-level overview of the services that a Product Security Incident Response Team might select at a point in time as part of their program. It may evolve over time as the mission or needs change and as the experience of the team grows. These referenced use cases cover a spectrum of developmental levels from a newly-established PSIRT, through a more advanced Incident Response team that has constantly refined its processes and added to their capabilities. As this evolution occurs, the team shifts from reactive operations to a more proactive mode of operating, reflecting greater maturity of their processes. The framework does not cover topics such as capacity - the number of vulnerabilities and incidents a team can manage simultaneously, or a maturity model such as SIM3[1] which indicates how well a team governs, documents, performs and measures their function(s).

Introduction:

So you’ve been told you need to have a PSIRT. Yay? Maybe up until lately you’ve had some ad-hoc processes, or people assigned as a secondary duty, or maybe you get the privilege to work for a brand new organization and build everything from the ground up? Whatever the case is, you’ve been tasked to assemble a team of people to help manage vulnerabilities identified in your products and offerings. PSIRTs come in a lot of different sizes and flavours, no two are exactly identical. At its core, a PSIRT takes in vulnerability reports, gives them some level of review and analysis, works with the appropriate parties

[1] https://opencsirt.org/maturity/sim3/

to craft security updates, and ultimately deliver those updates out to the organization's customers and partners.

This level seeks to describe the core services and functions a PSIRT needs to offer as it is starting out its journey into the world. Lessons written here are pulled from numerous organizations, much like yours perhaps, from a multitude of sizes, industries, sectors, and nations. We've all taken these first few steps you are about to undertake, and hopefully you'll benefit from where we predecessors may have once stumbled. Leveraging the Product Incident Response Teams Services Framework, we hope to highlight key areas to instruct a newly minted PSIRT on where their efforts will yield the best results. The use of the term "maturity" in this document is intended to describe a profile of a product security team and describe at a high level the capabilities the team provides their stakeholders.

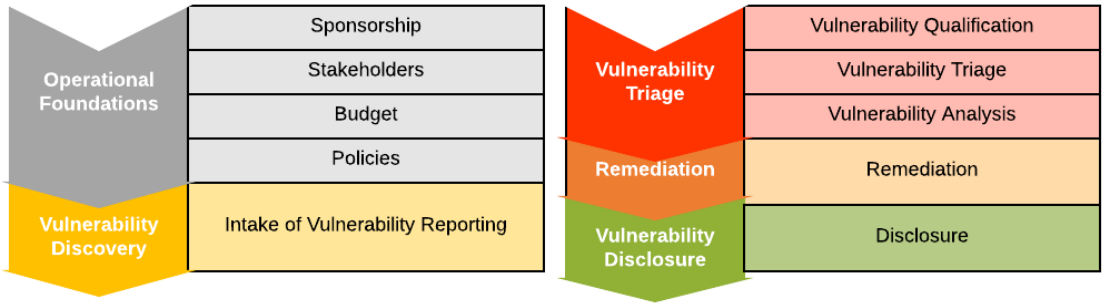

Figure 1: Listing of Maturity Level 1 desired Service Areas and Services

Getting Started - Operational Foundations

In the PSIRT Services Framework, we have a concept of Operational Foundations. This section identifies and describes the foundation of core components that an organization needs in order to plan, establish, and effectively operate a PSIRT.

> **50,000-Foot view -** In order to be effective, the PSIRT will need to have certain prerequisites before starting operations (things like budget, executive backing, equipment, etc.).

Think of the Operational Foundations like you might view the foundation of a building. These are concepts, processes, and personnel that must be in-place or planned for prior to starting to put the frame of the building up (your new PSIRT). Obviously, to be fully-functional and effective, you'll need to address each of the areas under the Foundations, but if you have limited time/resources or other constraints, there are a few that are essential. You'll want to ensure the PSIRT has some amount of dedicated staff that helps guide the organization through the complexities of managing vulnerabilities. The more focus they have, the better they will be able to serve your constituents.

Executive sponsorship

First and foremost: Why are you here? Why are you seeking to learn more about PSIRTs? At some point executive leadership decided that your company needed it. Attaining executive sponsorship/mandate is critical to everything else that follows. In your role on a PSIRT you're going to be asking people to do things they have not normally done or to prioritize your work over theirs to address some important security issue. Having the (written) mandate authorizing you to fulfill this role is critical. We combine

these two concepts for brevity, but know that in larger organizations there are nuances, but both concepts speak to having leaders support the PSIRT efforts.

It cannot be overstated that having a clear charter for the PSIRT and backing of organizational leaders is of importance. It will be challenging just starting out, but the PSIRT's job is exponentially simplified with the understanding, support, and backing of the organization's leaders.

Stakeholders

Ask yourself: "Who am I working for?" Stakeholders are the people you work with and work for. Each one will have different wants and needs from/of you, just as you will from/of them. The PSIRT is best served beginning with documenting their key stakeholders. Stakeholders are documented in more detail in the FIRST PSIRT Services Framework. By understanding who these groups are and what their requirements are the PSIRT can start to tailor itself to meet their desires and obligations. As the PSIRT matures and develops you'll begin to understand the unique difference between your stakeholder group and the need to customize reporting or communications to each.

Budget

Closely tied to leadership and stakeholder involvement, the PSIRT needs to have a budget to staff and provision itself. The size of the budget and number of resources the PSIRT will have varies widely across the PSIRT community, but ultimately the PSIRT should have sufficient funding to meet the business objectives of the organization. PSIRT leadership should be mindful that as the team and services grow, so will funding and staffing.

Policies and Procedures

So you have your high-level guidance from your company's leadership and their undying support. What next? You must have a documented set of rules that the PSIRT will be working with and enforcing. Just starting out you may only have one or two policies, but over time as the organization gains more experience the list could grow. Also, as you are just starting it is expected that you WON'T have many procedures documented...YET. This lack of documentation and process at this stage of the PSIRT's life may lead to inconsistencies in the reactive response. A key thing to do as the PSIRT starts working on procedures is to capture how it conducts itself and how it reacts to given circumstances. You can kickstart the PSIRT by borrowing existing good practices from project/program management, engineering, or support within your own company.

Some examples of internationally recognized standards that may be useful to PSIRTs starting out (or ones looking to close any gaps) are ISO/IEC 29147 Information technology -- Security techniques -- Vulnerability Disclosure and ISO/IEC 30111 Information technology -- Security techniques -- Vulnerability Handling Processes. In addition to reviewing these standards, PSIRTs could also establish the following policies:

- Vulnerability Management Policy (as covered in ISO30111)
- Information Handling Policy (as covered in ISO/IEC 29147)
- Vulnerability Scoring/Prioritization Policy
- Remediation Service Level Agreement
- Vulnerability Disclosure Policy (usually a public documentation)

Wherever possible, we'll provide links or examples to useful industry standards and other guidance in the annex of this document.

PSIRT Entrypoint - Vulnerability Discovery

Do you have a vulnerability if no one knows about it? With the foundation in place, the first step to solving a problem is knowing that you have a problem. In the PSIRT Services Framework we have the Vulnerability Discovery Service Area.

> **50,000-Foot View -** After you have some processes and people, you need to find things for them to do.

Having this capability allows the PSIRT to receive reports so that later action can be taken from them.

Intake of Vulnerability Reporting

There are many methods or paths this can take, but to begin work on remediating a security flaw, the PSIRT must first be made aware of the vulnerability. You probably do not proactively hunt yet for vulnerabilities or such reports yet, but rather you need to make sure others (internal and external) know how to reach you (e.g. publish your e-mail address) when they find a security issue. In addition to publishing your email address, it is also recommended to provide pgp key to receive vulnerability reports. It is also common to setup a web form for report intake.

Just starting out, the PSIRT may only be open for business for third-party researchers or internally reported vulnerabilities. You'll most likely want to closely work with this group to get more efficient in handling their communication and quickly get issues routed to the appropriate teams to correct. As you mature, you'll start to take on more and different finder types (for example - being able to take security bug reports from customers or your support or sales organizations).

The Next Step - Vulnerability Triage and Analysis

Vulnerability intake and triage commence the case management function of a PSIRT. While the order of operations is very similar among PSIRTs, there are variations within, like the exact point of when a 'case' is created or the personnel performing different functions within a case. Where organizations receive a high volume of vulnerability reports, they may consider performing initial triage to validate reports before cases are created. Vice versa, a case may be created before triage in organizations where the volume of vulnerability reports is low. The ultimate goal among PSIRT is to create an efficient and defined process.

Depending on what product and/or service and the technologies that comprise those items this stage can vary widely between PSIRTs. Hardware vendors might involve complex machines that go "bing" or leverage electrical or mechanical engineering practices, while a company that develops and releases software might use arrays of scanners or manual code reviews to better understand the problem.

> ***50,000-Foot View -*** *How big is the problem? Is it bigger than a breadbox? Does it even really affect you?*

Vulnerability Qualification

Once the PSIRT has received the report someone needs to validate that the report is valid. Did the reporter mistake their findings? Is the report actually a desired feature being misinterpreted? The PSIRT

must be able to view and understand what has been reported to them and make a decision whether to accept the issue as a security vulnerability or reject it for whatever qualifications they set out. Sometimes this may not be within the Spirit's pervue, and the qualification stage is completed by a Product Engineering Team. Ideally with all of the work put into the Organizational Foundations services up front will help to have these roles and responsibilities clearly documented, communicated, and understood by all involved parties.

Consider capturing key vulnerability information from the beginning to avoiding pain points.

Machine readable formats help or a format that captures key technical information

can help you understand what is written for decision making on the content, can avoid redundancy, and reduce time in the process.

Some key resources PSIRTs can use to help document and communicate vulnerabilities are:

- Common Vulnerability Reporting Framework (CVRF)
- Common Security Advisory Framework (CSAF)
- Vulnerability Description Ontology (VDO)
- Vulnerability Handling Process ISO 30111
- Common Vulnerability Enumeration (CVE)

Please refer to the annex for additional details.

Vulnerability Analysis

Now that the PSIRT has a valid security vulnerability, someone must dig into the problem, understand how the flaw works, how it is triggered, which versions of products are affected and what the consequences are when the vulnerability can be successfully exploited. The PSIRT may conduct the analysis, but oftentimes the bug is routed to a subject matter expert on the impacted product or offering for more in-depth review. Minimally, the PSIRT catalogues the issue, passes it off to someone that can authoritatively understand and review it, and the PSIRT tracks actions taken to ensure the report is addressed on some level (the risk needs to be remediated, mitigated, transferred, or accepted).

And one quick note before moving on about prioritization and scoring. The PSIRT must adopt a way for scoring from the beginning. Good practices suggest to categorize if something is a vulnerability or not by using some kind of scoring system for all incoming reports. Having your criteria documented ahead of time will help inform your actions. Ideally you should use the Common Vulnerability Scoring System (CVSS), but you can also use your own.

The main message here is choose your yardstick and then measure all things against it. If you do not use CVSS then you must have a good explanation for customers on why you think that your scoring system is better than CVSS.

Fixing the thing - Remediation

Wow! By this point you've sure done a lot! You've taken the report from the finder. You've validated it, in fact, as a security vulnerability. You've also helped either conduct or facilitate that the issue is fully explored and understood. The next step is to conduct a cost benefit analysis and assess options for

addressing the vulnerability. There are many possible options to consider. A few examples could be to create a code fix that completely eliminates the risk of the vulnerability, create a set of instructions that would limit the risk to the vulnerability or decide not to deliver a fix at all.

> **50,000-Foot View -** Now that you found some things that are broken you really should try and fix them.

Remediation

The most important output of the process is actually resolving the vulnerability. The PSIRT can track or facilitate issues that are resolved with some type of remediation or mitigation. Once the appropriate team addresses the issue, the PSIRT can then move on to its final mission shepherding the bug from inception to closure.

The Final Countdown -Vulnerability Disclosure

This is the last stage of the security vulnerabilities life. Now that the flaw has been addressed, the PSIRT helps communicate the updates and related materials out to the company's stakeholders (internal and external).

> **50,000-Foot View -** Since you went through all of that hard work to get the problems fixed, you probably should tell someone about it.

Disclosure

This is when the PSIRT notifies (or helps coordinate notification by the responsible party) the consumers of the product or offering. Disclosure can take many forms, but fundamentally consumers and interested parties receive some notice that the product or service is affected by a problem and are provided documentation on how to resolve or mitigate the vulnerability. Additionally, the finders that discovered and reported the vulnerability are acknowledged, giving them credit where it is due. This rapport and goodwill will come into play later.

It is an industry-wide good practice to ensure the researcher/finder gets appropriately credited for the discovery in the PSIRT's advisory. Acknowledgement helps the finder further their careers and build their reputation, and you recognizing their efforts generates goodwill toward you from them. Ideally this small investment in your text encourages them to come back to you responsibly again as they discover future issues! ISO 29147 provides good reference for Vulnerability Disclosure.

Conclusion

In a nutshell this is what a PSIRT does and the core steps that it takes to fulfill its mission. As stated earlier, the actual execution of all these steps can take many different forms depending on the organization, its size, age, etc. Once these basic elements are in place, a PSIRT may wish to explore how they can expand the quality and scope of their services. It is important to consistently be able to provide these services and functions prior to thinking about taking on more work and adding new capabilities. A more mature PSIRT may dabble with adding the capabilities that are hallmarks of an Intermediate PSIRT.

Maturity Level 2 (Intermediate) - I am reactive, but I've trained for

it!

Introduction:

After you've mastered a few things and responded successfully to some flaw reports, you'll start yearning to (or are told to) add more services to internal and external consumers of your services. There is a broad range of services a PSIRT could offer, but it is important to focus on the initiatives which benefit your business the most. Manufacturers have very different business concerns and risks they are trying to mitigate than a cloud-native start-up, as do the consumers of those types of offerings.

Overall, we view PSIRTs in this middle-stage of maturity as internally-focused. They are starting to manage things well, understand who to interact with to get things done, but don't have the mileage or the personnel perhaps to expand out into the wider community. It is expected that all previous services have been put into place and vetted. If you can't ingest a bug report successfully yet, you should revisit Maturity Level #1 and apply those practices. Hone your skills there before you move onto some of these newer techniques and services.

In this stage of development PSIRTs will typically have capabilities along the following lines:

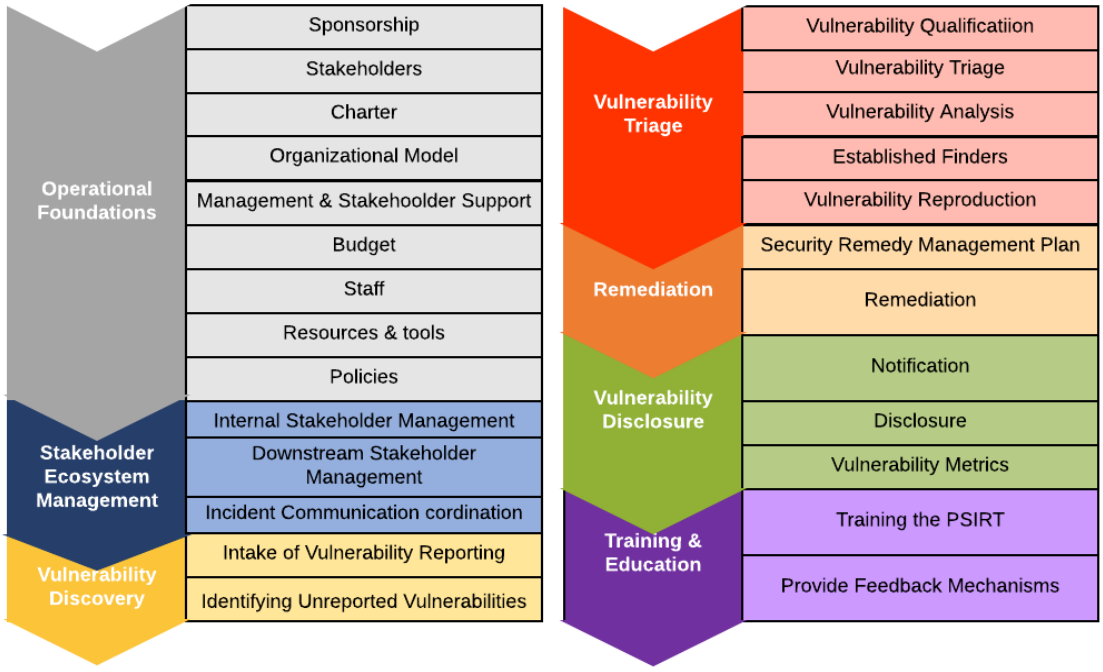

Figure 2: Listing of Maturity Level 2 desired Service Areas and Services

Back to Basics - Organizational Foundations

If you're thinking about "going to the next level", you should have successfully implemented everything from the Operational Foundations to some degree. These foundational steps establish practices, such as securing executive and management backing of the PSIRT. It is equally important to have critical items

such as policies, standards, and guidelines documented and funding for secured operations. When starting, some PSIRTs may elect to establish every single aspect, cutting some corners, but as the PSIRT grows and matures, these key things MUST be in place to ensure ongoing successful operations. The PSIRT Services Framework goes into detail about each area and function that should be in place.

(Not a) Communications Breakdown - Stakeholder Ecosystem Management

One of the single biggest areas that will separate an immature PSIRT from a more mature one is the understanding of the stakeholders who are involved with or concerned about the PSIRT's operation. These key services are really needed at this level. Organizations working on this level are expected to know whom they work with internally to review and react to vulnerabilities. The PSIRT has managed several vulnerabilities, and while there is still room for improvement, in general the PSIRT and its internal stakeholders know what to do when a vulnerability arises. The PSIRT should have a basic understand of who uses the company's products and services, as well as, in general, who supplies goods, code, or services to the organization.

PSIRTs functioning in this mid-level will have defined processes and channels to work through. Ideally you've moved beyond the basic firefighting, and you are able to recognize issues of grave importance to your products and to your customers. PSIRTs should have processes documented for when these major events arise, and be able to gather the appropriate stakeholders quickly to begin working on remediation. Another great thing is that ideally at this stage the PSIRT is regularly consulted by the members of the engineering/development teams of the organization. These interactions help ensure a near-real time feedback loop that ideally quickly identifies and corrects flaws.

Jinkies! A Clue! - Vulnerability Discovery

Now that the organization has gone through the vulnerability process several times, the PSIRT knows how to intake vulnerability reports. More mature PSIRTs will have multiple channels to collect vulnerability reports for your products/components and will have tooling to assist in the tracking of issues. At this level, the PSIRT and other internal engineers are discovering security vulnerabilities themselves (good job!). Being able to do this allows the organization to control the release of the updates, as opposed to working towards an externally-set deadline. This doesn't mean the PSIRT and their engineering partners can ignore these internally-found issues, but it allows them to schedule their resolution in a manner better fitting the organization release schedules or other resource availability instead of being handled as an emergency. Everyone likes fixing the thing, but also being able to go home on time and have dinner with the cat is nice too. Having opportunities to manage the release of internally-found flaws also gives all teams involved flexibility for when "the big one" is reported by a customer or security researcher.

It is important at this stage of maturity that the PSIRT invests time in collecting a comprehensive list of all the components that are included in a released product. This may have several names depending on the company but is commonly referred to as a Product Manifest or Bill of Materials (BOM). Having this list helps the PSIRT know what items to watch and test for vulnerabilities. As the PSIRT matures this list will help them understand what third-parties (such as open source projects) the PSIRT needs to interact with to remediate discovered vulnerabilities.

A word on SDLC

Software Development Lifecycle: SDLC, SDL, SSDLC, Security Engineering, Frank in QA, whatever your organization calls it, having connections into your company's SDLC process is important for the PSIRT. Each PSIRT's role and relationship to SDLC varies, but having open lines of communication and hooks into that process can help the PSIRT provide feedback and get security flaws corrected (ideally) prior to product release. Know whom to talk to, when products are phase-gated, and how best to provide input will benefit the PSIRT's mission or addressing discovered vulnerabilities.

Just the facts, ma'am - Vulnerability Triage and Analysis

The training wheels are off. Vulnerabilities are coming in from multiple sources, and the PSIRT has had practice digging into the reports and understanding if they are REALLY bugs or not (sometimes they aren't, it's lots of fun explaining that to an enthusiastic reporter keen on making a name for herself on the internet….good luck!). After practice, you now can qualify these reports and can manage them accordingly. Understanding that all bugs are not created equal, the PSIRT should consider having a quantifiable process for evaluating these incoming reports (sometimes referred to as a "Bug Bar").

Here is where the choice of how the PSIRT is structured comes into effect. The operating model you designed your processes and team around (Centralized, Distributed, Hybrid) will impact who is conducting vulnerability triage and analysis, and how that work procedurally is managed. Understanding WHO does WHAT with these incoming reports is critical to quickly reviewing and reacting to them. There are strengths and weakness to each model, and the PSIRT must understand what their part is in the workflow (Are you an active coordinator? Do you do the research? Do you just report the issues?).

By this time, you probably also have some "repeat offenders" reporting issues to you. That's great! The more you get to work with these folks, the happier EVERYONE is in the long run. It's always advisable to keep on friendly terms with finders as the internet can quickly make your life miserable if you're not getting along with a vulnerability reporter. Avoid that nonsense, treat them professionally and as quickly as your people, process, and tools allow. You probably have your finders recorded in a database that tracks some basic information about who they are, what they've found, and the quality of their reports.

It will also be helpful when working with finders to provide them updates on progress, typically done through a defect tracking or ticketing system. Providing tickets data back to the reporter helps show them you are appropriately addressing their concerns, creates a channel for collaboration with that finder, and ultimately should reduce the risk they will become frustrated and disclose the flaw before any agreed upon date.

This maturity allows you to better work with all reporters, coaching them into what makes an easily actionable report, and what is a hot mess that will take lots of back-and-forth on to get resolved (which frustrates EVERYONE). Remember, the more you do it, the better you get, the faster you can do it, the more time you're freeing up to improve yourself in other areas. At this stage of your development, the PSIRT or your engineering partners will have developed some form of reproduction capability. It's important to test out these flaws in an isolated place (to avoid ruining everyone's day on the network). Have documented processes of how to handle the exploit, how it will be tested, and protected from getting out into the wrong hands. Have a plan for how you might safely share reproducers internally (and possibly with external partners/stakeholders).

We mentioned CVSS scoring concepts for beginning PSIRTs. At this stage the team is proficient in evaluating the security vulnerability and should be able to describe security vulnerabilities using

industry-standard terms. Some PSIRTs may have delegated out the initial scoring to engineering partners (sometimes referred to as "security champions") with the PSIRT providing oversight and guidance as needed. Curating this network of security-minded partners across your organization can certainly help the PSIRT be more effective. Some PSIRTs may enhance their scoring by labeling issues with Common Weakness Enumeration (CWE). They also may create tooling to do automatic scoring and descriptions. Additionally, the PSIRT may also desire to provide additional context around the flaw from the perspective of how the organization's products are configured and deployed along with the CVSS score and define severity labels.

Remediation

As you get better at "doing PSIRT", you'll improve your tooling, documentation, and people's comfort levels around dealing with problems. You're now able to do more than just build a patch or fix, you can do it in a repeatable fashion, leveraging your current build processes. Maybe sometimes you do it a bit faster than normal, but you are no longer "recreating the wheel" and building each and every fix one-off and by hand. There's automation, it's repeatable, and while it's an emergency, people understand there is a process to handle it. Hopefully you've shared this process with your consumers, so that they get notification and mechanisms to get the updates quickly. A PSIRT working at this level should have policies, standards, guidelines, and processes documented; you might not account for everything yet, but the most common scenarios in your environment are accounted for. You also probably have enough policy by this point that you may need to have an exception process for things that will deviate from the rules.

I promise to tell the truth, the whole truth… - Vulnerability Disclosure

Your first few security vulnerabilities might have been new and scary, but now you've done several and your team is trained on what to do. You are doing more at this stage than just simple notification. Users of your things know about the updates as they are released, but now you're also doing a better job at coordinating with others. A good reference for this coordination can be found in the FIRST's Multi-Party Coordination and Disclosure Guidelines.

You're dipping your toe into the wider ocean of the ecosystem around you, and perhaps you're synchronizing your updates with other organizations to help limit risks to shared consumers. The PSIRT has defined circumstances that require advance customer communication before disclosure. They may have introduced email subscriptions and notifications for when disclosure is posted. They have matured in cross industry communication and internal communication to sales and support teams before disclosure. By now you also have been tracking metrics on yourself and your delivery of security updates. The PSIRT Services Framework has suggestions on many metrics that could be tracked throughout each of these phases. You feed this back into your processes after you analyze your performance and you improve how you're managing these incidents. PSIRT may consider The Traffic Light Protocol (TLP) for information sharing.

Training

This has been the secret sauce that sets you apart from beginning PSIRTs: experience and reflection. You've taken what you've learned, applied it and made yourself better. Ideally you've written this new wisdom down so that you can share with others within your organization. You may have created educational training specifically for certain stakeholders such as public relations, legal, and executives so

that they know their roles and what you expect and need from them. As you are able to add new associates to your team or extended team you can provide them consistent training, so that everyone knows the same information and can act consistently with your practices and policies.

FIRST has put together a series of training videos that can help educate new PSIRTs and those seeking to get better at their craft.

Conclusion

Every vulnerability is a learning opportunity for the PSIRT. By executing their workflow, the PSIRT will gradually find areas where practices or tools can be optimized. Each end-user interaction helps coach the PSIRT into how to adjust the overall deliverables of the organization to best meet those needs. As the PSIRT begins to execute consistently on these techniques just discussed, they'll develop efficiencies and start to have more time to focus on improving their situation. Given enough time, practice and reflection, the PSIRT will begin to move to a more advanced stage of performance.

Maturity Level 3 (Advanced) - Proactive...we're ready for anything (mostly)

Introduction

Congratulations! You've been PSIRT'ing a while now, solved lots of problems and have really embedded yourself into the fabric of your organization! Pretty cool, huh? Your customers are (generally) happy and serviced well by your management of issues that impact them. You feel unstoppable! So… what's next? You've got the basics down, so now we'll dive into how to make adjustments and improvements to start doing things like some of the elite PSIRTs out there.

To get to this level it is understood that your process for ingestion, analysis, and delivery of security updates for your products is well-understood within your organization. Your internal peers know who you are and what the PSIRT does and you've all weathered many cycles of issues to the point that the process and requirements are not new or surprising to anyone you interact with internally. You've also survived several large vulnerabilities or an attack that is larger than a "typical" problem. These larger events have helped identify areas where your processes and documents needed improvement.

You've deployed enough tooling, process, and people that you can keep pace with managing incoming new work and following up on issues that are in-flight. You're now able to manage both the daily "business as usual" vulnerability management issues and also these larger, more consuming problems as well.

PSIRTs operating at this level are exemplified by proactive and consistent behaviors. You no longer just let things happen to your organization and products, you're now taking PROACTIVE steps to seek problems and people out. You've been through enough incidents that now you're also taking steps to predict when things could occur. Maybe you have some cool tool and you're basing actions on historic events to predict what could happen? Maybe you are very effectively managing risks and understand

what types of behavior could impact your business… and you're working to mitigate them before they become problems. The capabilities PSIRTs at this level perform speak to a well-managed, adaptive process.

A listing of services a team at this level would be delivering could look like this:

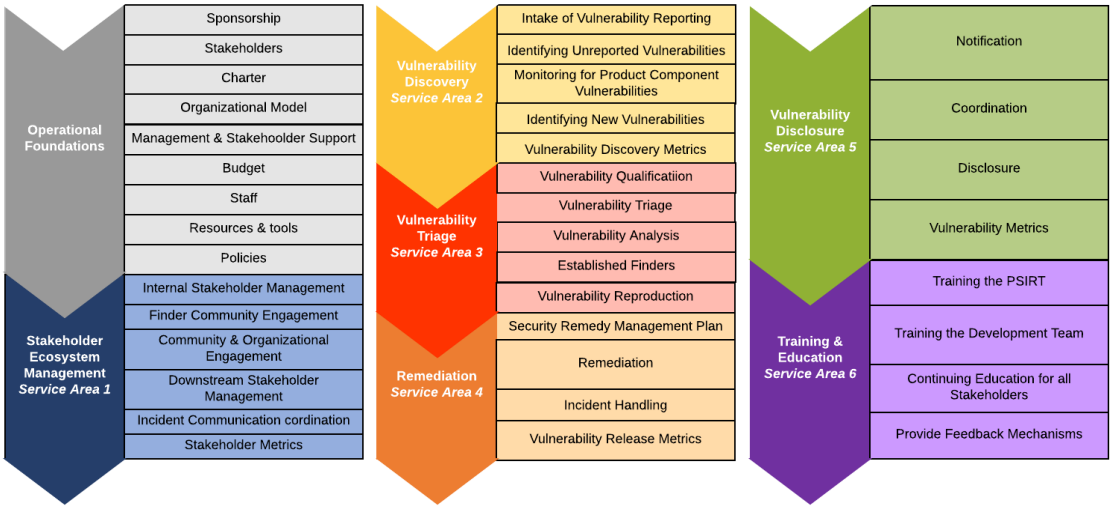

Figure 3: Listing of Maturity Level 3 desired Service Areas and Services

We'll dive deeper into the capabilities that a seasoned PSIRT exhibits.

Rock-solid Foundations

To consider yourself mature, all of these processes and services need to be long-established. Each of these services are vital to the operation, funding, and direction of the PSIRT. If you're missing things here, go back and reflect upon what you might not have gotten consistently down or that you may be missing. Acquire internal support to get these things established. You must be this tall to ride the ride. Come back and read further once you've done all of this.

Putting the Stake into Stakeholder...mmmm...steak

PSIRTs at this advanced stage operate with confidence. That confidence is partially developed by frequent and active conversations with parties involved in the products and processes of the company (and looking outside too). By this time, the PSIRT has invested in developing open, honest relationships with all involved stakeholders...the list is long. At this stage of evolution the PSIRT has crafted standardized communications, leveraging playbooks and templates, and stakeholders will have multiple avenues to both provide and receive feedback on the process. The PSIRT should participate in regular product program meetings that are core to the organization, and should be well-versed in the upcoming release schedule.

As with all portions of the Framework the PSIRT uses at this level, metrics and reporting are key for continued success. The PSIRT and its stakeholders will have clearly defined objectives and measurements of success. So Key Productivity Indicators could be things like "Customer Satisfaction",

Net Promoter Score, Security Vulnerability Patch Delivery, Service Level Objectives/Agreements, or the like that are reviewed by the PSIRT and its stakeholders to ensure continued success of the operation. These metrics help inform the PSIRT to make positive business choices and influence internal behavior of all participants in the remediation process.

A key hallmark of PSIRTs operating at this level is a robust understanding and engagement with external stakeholders. Whether it has been helping establish vulnerability response mechanisms and processes, integrating or overseeing some form of Secure Software Development Lifecycle, or even simply reacting to the internal support of sales organizations. Now that the PSIRT has a strong foundation, they will begin to shift from being mostly internally-focused to engaging more with external parties.

The PSIRT should seek to align better with industry peers, or security researchers; perhaps outreach programs of some form are created to help build strong relationships. Of most benefit would be building relationships with external "upstream" providers, and fully understanding how they react to security issues, which in turn would cause impact to goods and services the PSIRT helps govern. This third-party component management (like everything the framework details) could take many different forms.

Discovering the Unknown (or at least it wasn't known to you to start out with)

As PSIRTs enter this phase of their existence they are becoming masters of their own domain. Good relations with internal stakeholders give the PSIRT better insight into the release pipeline. They will be aware of upcoming features/functions/packages and be better prepared for the future. The PSIRT has either developed or acquired tooling to manage the influx of vulnerability reports from multiple sources. Ideally at this stage of evolution the PSIRT is actively assisting in the hunt for product vulnerabilities. This behavior has become integral to the development and maintenance process.

The PSIRT has influenced the organization to conduct better vulnerability analysis and scanning throughout the offering development process so that more security bugs are caught prior to product launch and are remediated before an end-user ever could be affected by them. As issues get reported, each needs to be analyzed to see if additional variants might exist (those researchers are smart, but they don't always know everything). The detailed product knowledge the PSIRT and the Product Engineering teams have may enable them to discover additional ways to exploit. The PSIRT is managing security risks in third-party components and actively monitoring sources for vulnerability reports (social media, news outlets, conference papers, etc.). These can be vital early indicators of future attacks or classes of vulnerabilities not yet discovered by the organization.

What's the Diagnosis, Doc?

The team has been together for a while now and understands the product landscape they help oversee. The PSIRT has developed processes to quickly and accurately assess vulnerability reports, hopefully leveraging previous finders or a better understanding of how the organization's products could be exploited. Experience and reflection help provide efficiencies in process and tooling so that the team can continue to reflect on their performance and seek to improve offerings.

Having reacted to reported flaws before, the PSIRT has insights into many security researchers and reporters in their particular area. Some of these researchers have proven themselves as extremely qualified. They may get priority flagging and get moved straight into the analysis phase, bypassing

remedial triage steps that unproven reporters may still have to hurdle. The PSIRT should have strong feelings for what makes a "good" report and what elements/data points help them more quickly validate (or disprove) a researcher's findings.

The PSIRT also should have a process to route issues into some form of secure reproduction system to recreate attacks and vulnerabilities. This allows the team to play out "what if" scenarios.

The PSIRT can consult with development teams and help provide feedback to avoid common coding mistakes and security flaws. Tools like CWE allow engineers to look at historic flaws to help avoid future behaviors. This guidance helps to avoid costly fixes after a release by integrating that feedback early into a development lifecycle.

Fixing the Thing

Ideally by this stage, with repeatedly fixing previous issues, the PSIRT and all stakeholder participants are now able to deliver updates consistently. All involved parties know what to do, they have clear expectations of the timeline to deliver the mitigation, and have sufficient resources to devote to correcting the problem. As a technical issue is being corrected, the PSIRT is ensuring that appropriate documentation and communications are being developed so that as the issue is reported to the public, partners, peers, and consumers can all understand the issue, how it was fixed, and if there were any alternatives to "just patching".

Delivery of the update is a routine process, not one newly minted in the heat of the moment. The final delivery may take the form of a standardized update release window, it might be prepared to go "over the air" via automated updating mechanisms, the problem and product involved will help shape which options the organization has available to deliver the mitigation to end-users that are affected by the problem. Whatever the end-user must do to react to the vulnerability is prepared and waiting for the time that it is publicly disclosed.

Hark! I bring tidings of …. broken stuff (but I'm also going to tell you how to fix it!)

The PSIRT should ensure that all stakeholders are alerted when the vulnerability is ready to be made public and updates are available. Thinking back to the earlier Stakeholder Management section, the PSIRT understands the various groups that need to be informed and how best to engage with each. When the time comes to release updates or a public statement, the PSIRT knows which channels are used to inform each constituent appropriately.

In scenarios where the PSIRT is either reliant upon an upstream or third-party provider, or they themselves provide the products or services downstream, it is important that the PSIRT understands how best to inform each of these different groups. If the PSIRT is part of some larger ecosystem, where multiple vendors are impacted by the same bug, the involved PSIRTs will typically coordinate a mutually agreeable time ("embargo period") where all impacted customers can be simultaneously informed. Ideally all of the various updates are released at the same time so that no one end-user group is adversely affected over another, with a goal of minimizing the period of time where malicious actors can take advantage of the now-public vulnerability.

Teach me, Sensei

PSIRTs at this stage of evolution have been involved with various levels of training up to this point. They may have taken tactical training on a product or security technique or documented standard operational procedures into playbooks and then worked with new associates to understand them.

Again, PSIRTs at this level are more proactive than they used to be. The PSIRT can either participate in or possibly deliver secure development training content to their internal peers or other stakeholders (depending on resources and mission). They augment training and documentation to reflect good practices and learnings, helping build the “next generation” of PSIRT associate or partner. The PSIRT is exponentially stronger if its developers, engineers, and product teams understand secure coding, privacy, and information security techniques, and can participate in executing on those principles.

Conclusion

Securing your organization's products and services is a journey, not a destination. Every day the threat landscape shifts, new technologies are created, new ways of thinking are developed, new threats arise, and old ones stagnate. Hopefully these maturity levels have helped inform you as you embark upon your own PSIRT journey!

Beyond “Level 3”

What? …. Sounds Impossible…. but it is NOT!

Annex 1: Supporting Resources

For a complete list of supporting resources and glossary refer to the PSIRT Services Framework https://www.first.org/education/FIRST_PSIRT_Services_Framework_v1.0.pdf.

Annex 2: Illustrations

Annex 3: PSIRT Charter

PSIRTs should have a defined charter that describes what the PSIRT does, and its scope. Most charters have the following items:

- ❑ **Mission Statement**

  The mission statement defines the team's purpose and activities (see examples in Annex 3).

- ❑ **Stakeholders**

  Stakeholders are the parties the PSIRT serve. Review Service 1.1: Internal Stakeholders in the PSIRT Framework for more information.

- ❑ **Affiliation and sponsoring organization**

  The sponsoring organization as defined in the Executive Sponsorship supports the PSIRT goals, actions, and provides resources for its operations.

- ❑ **Scope**

  As stated in the PSIRT frameworks, PSIRTs are as unique and varied as the products they help protect, their stakeholders and their organizational structure. Scope speaks to the responsibility and influence granted to the PSIRT team across the entire organization.

Annex 4: Examples of Mission Statements

- **Microsoft**
  Microsoft Security Response Center (MSRC) coordinates and mitigates security issues impacting Microsoft's customers, brand and systems. MSRC is the liaison between external security researchers and Microsoft product teams. MSRC provides a critical interface, coordinating with security researchers and Microsoft product teams to document and fix reported security vulnerabilities in Microsoft products.

- **IBM**
  The IBM Product Security Incident Response Team (PSIRT) is a global team that manages the receipt, investigation and internal coordination of security vulnerability information related to IBM offerings. IBM PSIRT is a focal point for security researchers, industry groups, government organizations, and vendors to report potential IBM product security vulnerabilities. This team will coordinate with IBM product and solutions teams to investigate, and if needed, identify the appropriate response plan. Customers of IBM offerings should continue to report all product related issues, including potential security vulnerabilities, to IBM Technical Support. Maintaining communication between all involved parties, both internal and external, is a key component of our vulnerability response process.

- **Brocade**
  Brocade Product Security Incident Response Team's mission is to protect Brocade and its customers by managing the receipt, investigation and coordination of security vulnerability that affects the confidentiality, integrity or availability of Brocade products and services.

- **DELL EMC**
  Dell strives to help our customers minimize risk associated with security vulnerabilities in our products. Our goal is to provide customers with timely information, guidance and mitigation options to address vulnerabilities. The Dell Product Security Incident Response Team (Dell PSIRT) is chartered and responsible for coordinating the response and disclosure for all product vulnerabilities that are reported to Dell.

- **Red Hat Product Security**
  To help protect customers from meaningful security concerns in Red Hat's products, services, and projects; by ensuring our products are secure, vulnerabilities are investigated, and issues that matter are fixed.

  Provide a superior quality customer experience of product security through clear, accurate, timely, and authoritative information. Ensuring our value is recognized as an important part of the value of subscription.

To build and maintain a cohesive team of highly successful, passionate and happy associates who work effectively and are regarded internally and externally as leaders in security.

- **Honeywell**

  The Honeywell Product Security Incident Response Team (PSIRT) manages security vulnerabilities and incidents for all Honeywell products, services, and components. The PSIRT focuses on the identification, assessment, mitigation, and disposition of the risks associated with security incidents and vulnerabilities of Honeywell products. This includes offerings, solutions, components and services. The PSIRT services are an integral part of the Secure Development Lifecycle (SDL).

Annex 5: Charter Template

A charter should include a purpose, business problem, background (optional), teams charter and the main sponsor. For example:

> *The Product Security Incident Response Team (PSIRT) supports development teams with all security related aspects of the company's products. This includes but is not limited to identification, mitigation and disclosure of vulnerabilities that affect the supported products, offerings and solutions developed, sold or distributed by the company. PSIRT is sponsored and funded by the [Insert appropriate executive here] for the company.*

Annex 6: Policy Template

Policies clearly communicate "what" is expected of employees with respect to product security vulnerabilities. This is different from processes which explain "how" employees would meet the published policies. Each policy or set of policies are unique to the company and security culture. Key ingredients of a policy document could also include responsible executive, responsible office, effective date, last update, who is affected by the policy and the actual policy. Some potential examples of what may be included in the policy are as follows:

- *All known security vulnerabilities must be submitted to the Product Security Incident Response Team (PSIRT) for processing and disposition.*
- *Security vulnerabilities must be triaged within [#] days of receiving a report with the final result being a statement of impact to the company's supported products and customers.*
- *The company will deliver remediations for reported security vulnerabilities based on the impact to the company's products and as defined by PSIRT.*
- *The company will publicly disclose security vulnerabilities only when a remediation is available and customer action is required.*

For further details please revisit Maturity Level 1 policies and procedures (note Vulnerability Disclosure Policy is usually a public documentation-ISO/IEC 29147).

Annex 7: Sample Checklist

Maturity Level 1

- ❑ Operational Foundations
  - ❑ Secure executive sponsorship
  - ❑ Identify stakeholders
  - ❑ Establish budget
  - ❑ Establish policies
- ❑ Vulnerability Discovery
  - ❑ Intake of vulnerability reporting
  - ❑ Establish PSIRT email address with PGP key
- ❑ Vulnerability Triage
  - ❑ Define vulnerability intake process
  - ❑ Establish internal workflows for vulnerability qualification, triage, and analysis
- ❑ Remediation
  - ❑ Analyze remediation options
  - ❑ Document the fix or mitigation
- ❑ Vulnerability Disclosure
  - ❑ Adopt industry standards such as CVE/CVSS to standardize how you will document and communicate/disclose vulnerabilities
  - ❑ Create templates for communications
  - ❑ Communicate to stakeholders
  - ❑ Acknowledge researchers/finders

Maturity Level 2

- ❑ Operational Foundations
  - ❑ Establish charter
  - ❑ Build organizational model
  - ❑ Ensure management & stakeholder support
  - ❑ Identify additional staffing requirements
  - ❑ Identify additional resources and tools
  - ❑ Ensure branch/version support policies and lifecycles are understood
  - ❑ Create baseline metrics
  - ❑ Establish a product registry with dependency mapping
- ❑ Stakeholder Ecosystem Management
  - ❑ Identify internal stakeholders who will be key to vulnerability management
  - ❑ Identify downstream stakeholders
  - ❑ Establish incident communications and coordination
- ❑ Vulnerability Discovery

- ❑ Establish process for discovering unreported vulnerabilities
- ❑ Vulnerability Triage
  - ❑ Identify finders who repeatedly come to you with quality reports
  - ❑ Develop internal vulnerability reproduction capability
- ❑ Remediation
  - ❑ Formalize security remedy management plan
- ❑ Vulnerability Disclosure
  - ❑ Create system to notify stakeholders
  - ❑ Establish vulnerability metrics
- ❑ Training & Education
  - ❑ Provide training to the PSIRT team members
  - ❑ Provide feedback mechanisms

Maturity Level 3

- ❑ Operational Foundations
  - ❑ Build out organizational policies
  - ❑ Determine cost of vulnerability management
- ❑ Stakeholder Ecosystem Management
  - ❑ Begin direct engagement with the finder community
  - ❑ Community & organizational engagement (i.e. join organizations such as FIRST.org)
  - ❑ Create stakeholder metrics
- ❑ Vulnerability Discovery
  - ❑ Monitor for product component vulnerabilities
  - ❑ Identify new vulnerabilities (i.e. monitor feeds, forums, and external sites)
  - ❑ Establish vulnerability discovery metrics
- ❑ Remediation
  - ❑ Define advanced incident handling
  - ❑ Develop vulnerability release metrics
- ❑ Vulnerability Disclosure
  - ❑ Build playbook for stakeholder/industry coordination
- ❑ Training & Education
  - ❑ Train development teams
  - ❑ Establish continuing education of all stakeholders