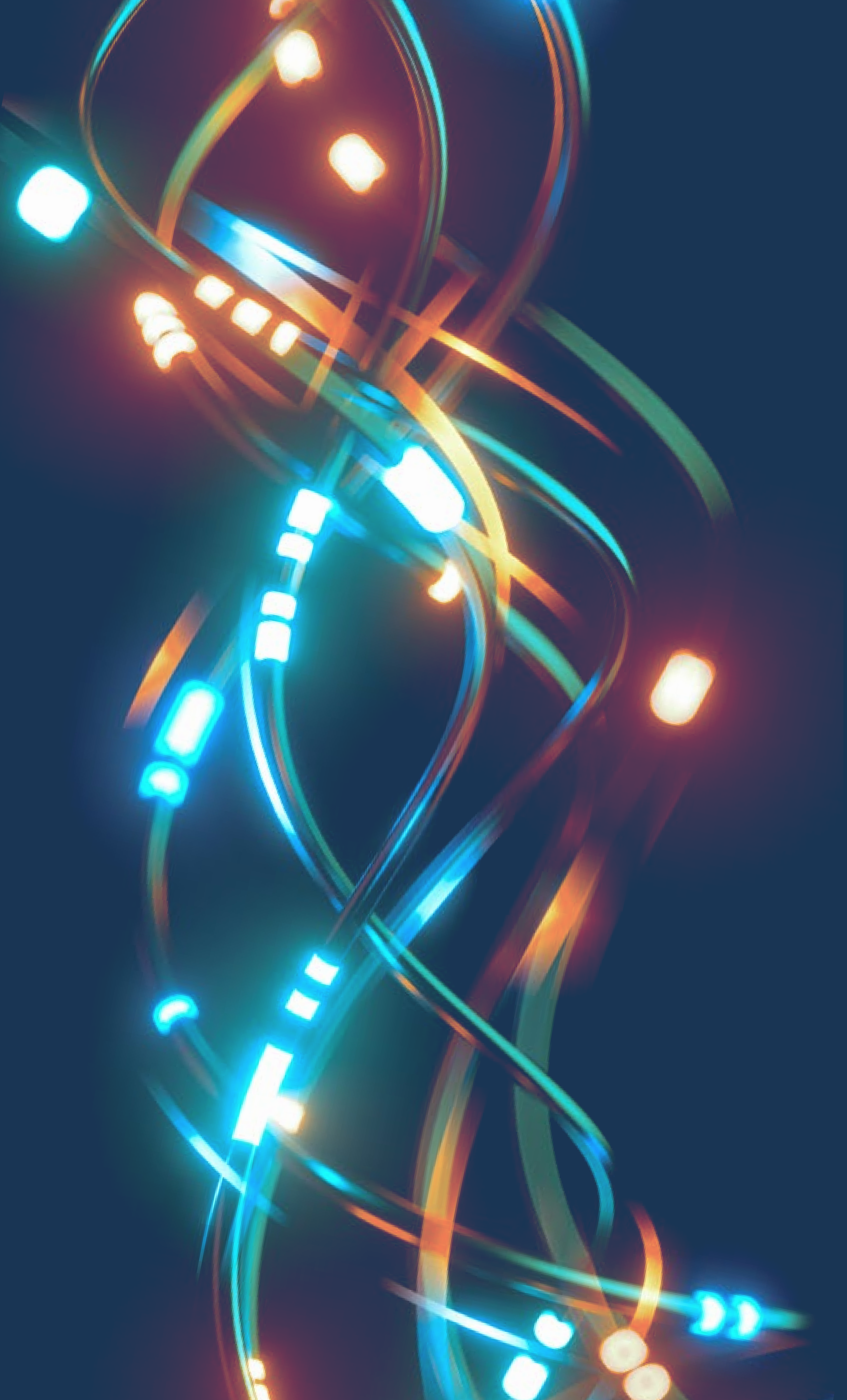
PASTA

Process for Attack Simulation & Threat Analysis

VERSPRITE



CONTENTS

OVERVIEW

Process for Attack Simulation and Threat Analysis (PASTA) is a threat modeling methodology, co-developed by VerSprite's CEO Tony UcedaVelez. It provides a process for simulating attacks to applications, analyzing cyberthreats that originate them, and mitigating cybercrime risks that these attacks and threats pose to organizations. The process is employed by security professionals across industries to prioritize risks and develop a mature cybersecurity framework that is woven into the business culture and the application development process.

PASTA consists of seven stages aimed to discover and minimize risks and associated impact. By following this process, businesses can determine the adequate level of countermeasures that can be deployed to mitigate the risk from cyberthreats and attacks to applications.

SOLVENCY

PASTA is a strategic process that provides organizations with practical steps to effectively countermeasure and mitigate existing vulnerabilities, analyze attacks that can exploit these vulnerabilities, and map these attacks to threat scenarios that specifically focus on applications as business-asset targets.

PASTA METHODOLOGY HELPS ORGANIZATIONS WITH:

- Defining a risk mitigation strategy
- Improving application security
- Building security into the Software Development Life Cycle (SDLC)
- Identifying application vulnerabilities and design flaws
- Analyzing application security risks
- Developing time and cost-efficient mature security frameworke security framework

COMPARATIVE ANALYSIS

Most formal threat modeling tools and methods, that have been developed today, are either software-centric (modeling threats to application software) or data-centric (analyzing risk that threat pose to data assets). These methods and tools model independently of the business context, objectives, and specific impact.

Limited to technical impact analysis, these methods fail to model the threat agents' motives and their drivers for attacking businesses.

PASTA methodology was developed to take into consideration and determine both technical and business risks. It allows an in-depth and focused analysis of the cyberattacks and attack scenarios, that are simulated to determine the possible exploits, and devising countermeasures to eliminate them.

WHO BENEFITS FROM PASTA METHODOLOGY

PASTA provides essential tool and guidance for organizations prioritizing the security of their assets and working to minimize the risks and impact. It helps all stakeholders involved in the assessing threats develop effective risk mitigation strategies.

PASTA methodology is wildly used by Architects, Developers, Security Testers, CISOs, Business Managers, Project Managers, and Information Risk Officers.

BENEFITS BREAKDOWN

VerSprite's PASTA threat modeling methodology benefits an organization across all vectors.

The methodology allows **Architects** to estimate how vulnerabilities to the application affect threat mitigation, identify trust boundaries and classification of the data assets, and apply countermeasures through proper design.

PASTA helps **Developers** understand which components of the application are vulnerable and learn how to mitigate the vulnerabilities.

Security Testers can use security requirements derived through the methodology to create positive and negative test cases.

Project Managers can effectively prioritize remediation of security defects according to the risks.

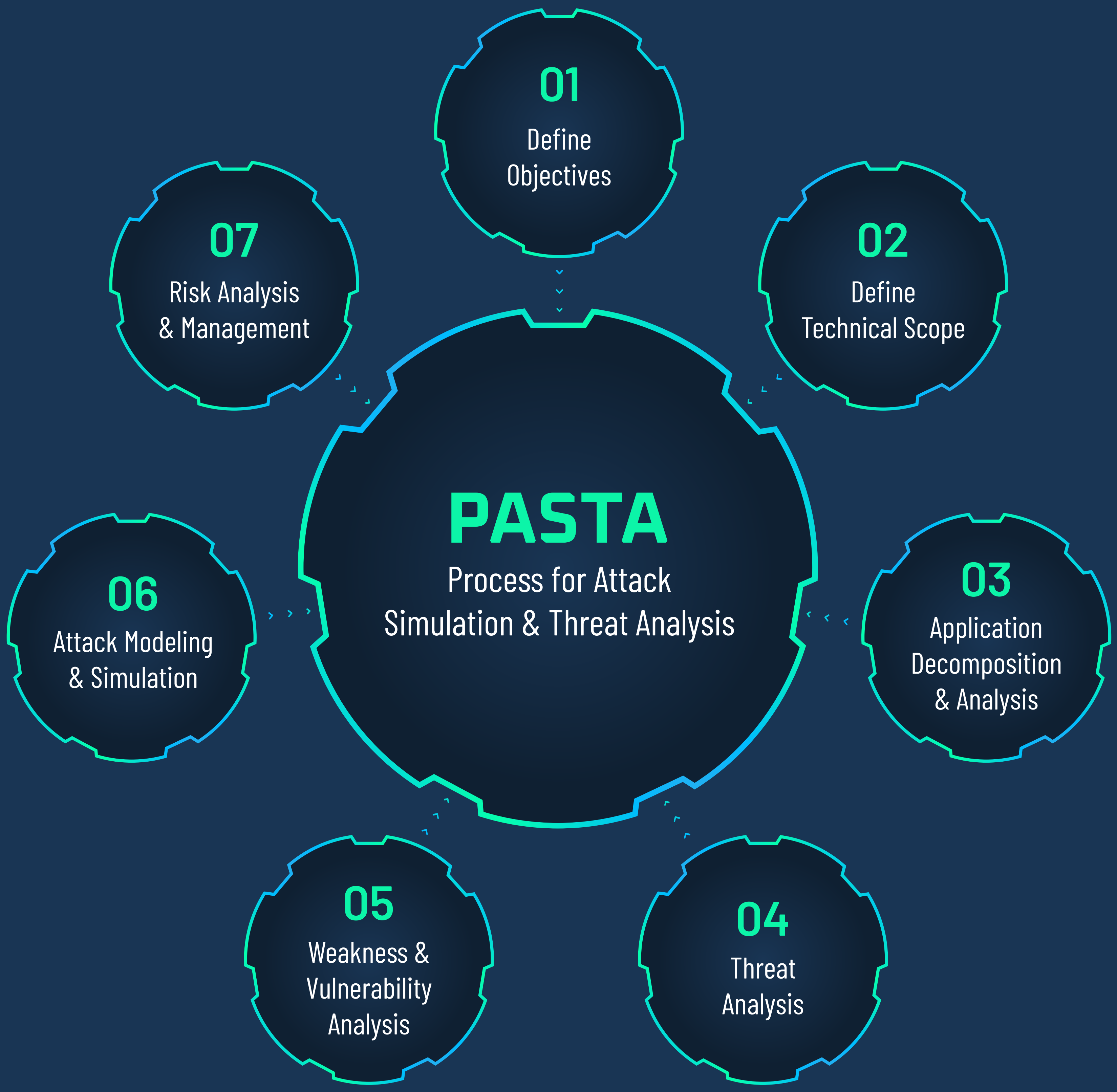
Business Managers can determine which business objectives have impact on security.

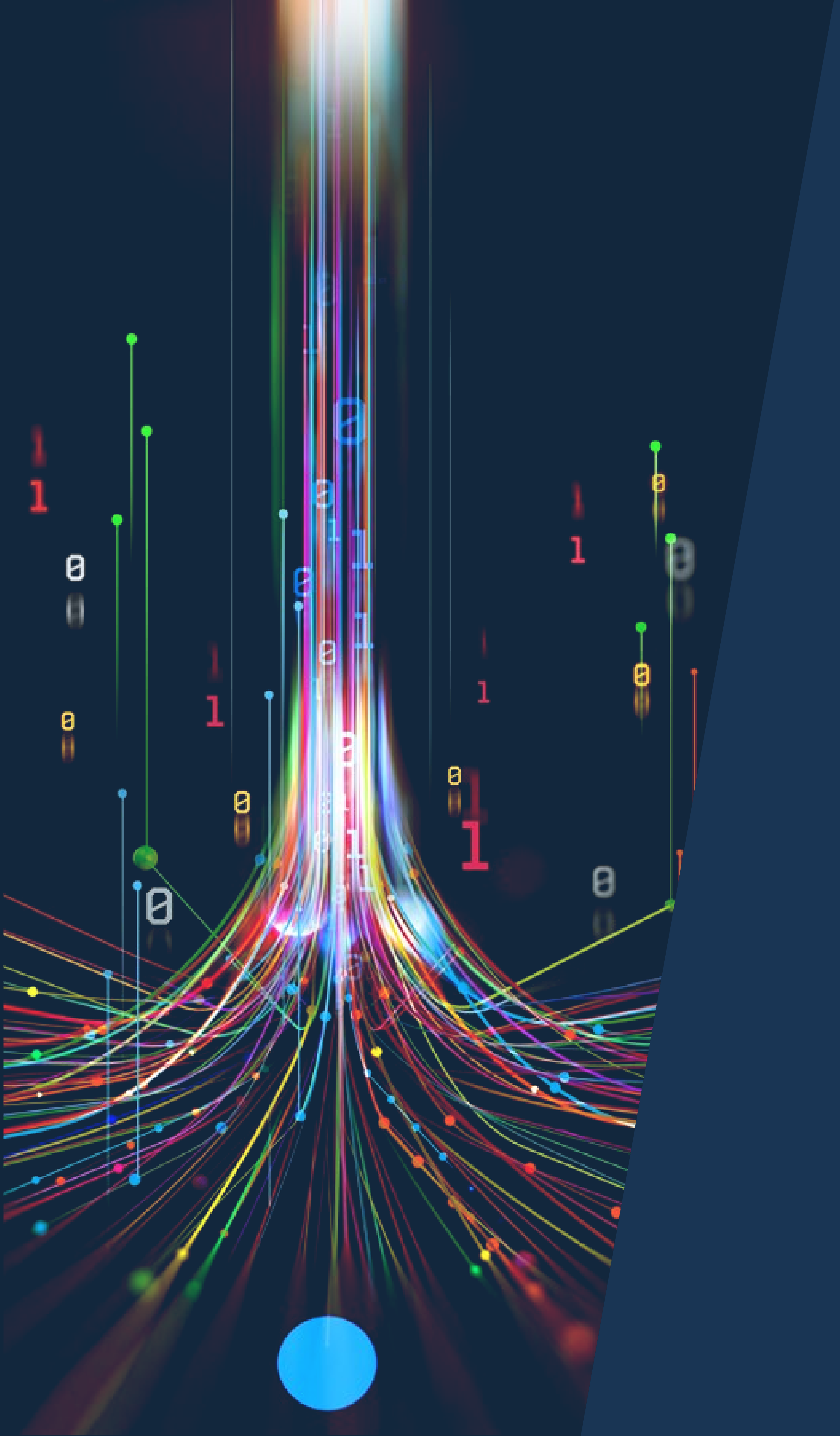
The methodology helps **Information Risk Officers** make strategic risk management decisions by mitigating technical risks while considering costs of countermeasures against the costs associated with business impact.

PASTA is a framework for modeling threats to the application environment. Through the seven-stage comprehensive process, the methodology looks at the business as a whole from both insider and threat agents' points, taking into consideration business objectives, application implementation and SDLC, network, users, compliance and security measures and tools.

It provides an opportunity to mitigate any business risks that have been identified and qualified as a part of the threat modeling effort.

STAGES OF PASTA METHODOLOGY


PASTA
Process for Attack Simulation & Threat Analysis
01 Define Objectives
02 Define Technical Scope
03 Application Decomposition & Analysis
04 Threat Analysis
05 Weakness & Vulnerability Analysis
06 Attack Modeling & Simulation
07 Risk Analysis & Management



STAGE I: DEFINING THE OBJECTIVES

Determining business objectives and ensuring an appropriate level of security requirements to support the business goals for the application, while meeting security standards compliance.

STAGE II: DEFINING THE TECHNICAL SCOPE

Defining the technical scope and boundaries of threat modeling. Key factors – various technologies, software and hardware, components and services used by the application.

Categorizing any architectural and technology components, which function is to provide security controls (e.g., authentication, encryption) and security features (such as protection of CIA).

STAGE III: APPLICATION DECOMPOSITION & ANALYSIS

Decomposing the application into essential elements of the application architecture (users, servers, data assets, etc.) which can be further analyzed for attack simulation and threat analysis from both the attacker and the defender perspective.

STAGE IV: THREAT ANALYSIS

Enumerating the possible threats targeting the application as an asset.

Identifying the most probable attack scenarios based upon threat agent models, security event monitoring, fraud mapping, and threat intelligence reports.

The final goal of the stage is to analyze the threat and attack scenarios that are most probable, and prioritizing them for the attack simulation.

STAGE V: VULNERABILITY & WEAKNESS ANALYSIS

The main goal of this stage of the methodology is to map vulnerabilities identified for different assets that include the application as well as the application infrastructure to the threats and the attack scenarios, which were identified in the previous stage.

Formal methods for mapping threats and vulnerabilities, such as threat trees, are employed for determining the ones to use for attacking the application assets.

STAGE VI: ATTACK MODELING & SIMULATION

This stage analyzes how the application and application context (user-agents, environment) can be attacked by exploiting vulnerabilities and using various attack libraries and attack vectors. Formal methods include attack surface analysis, attack trees, and attack libraries-patterns.

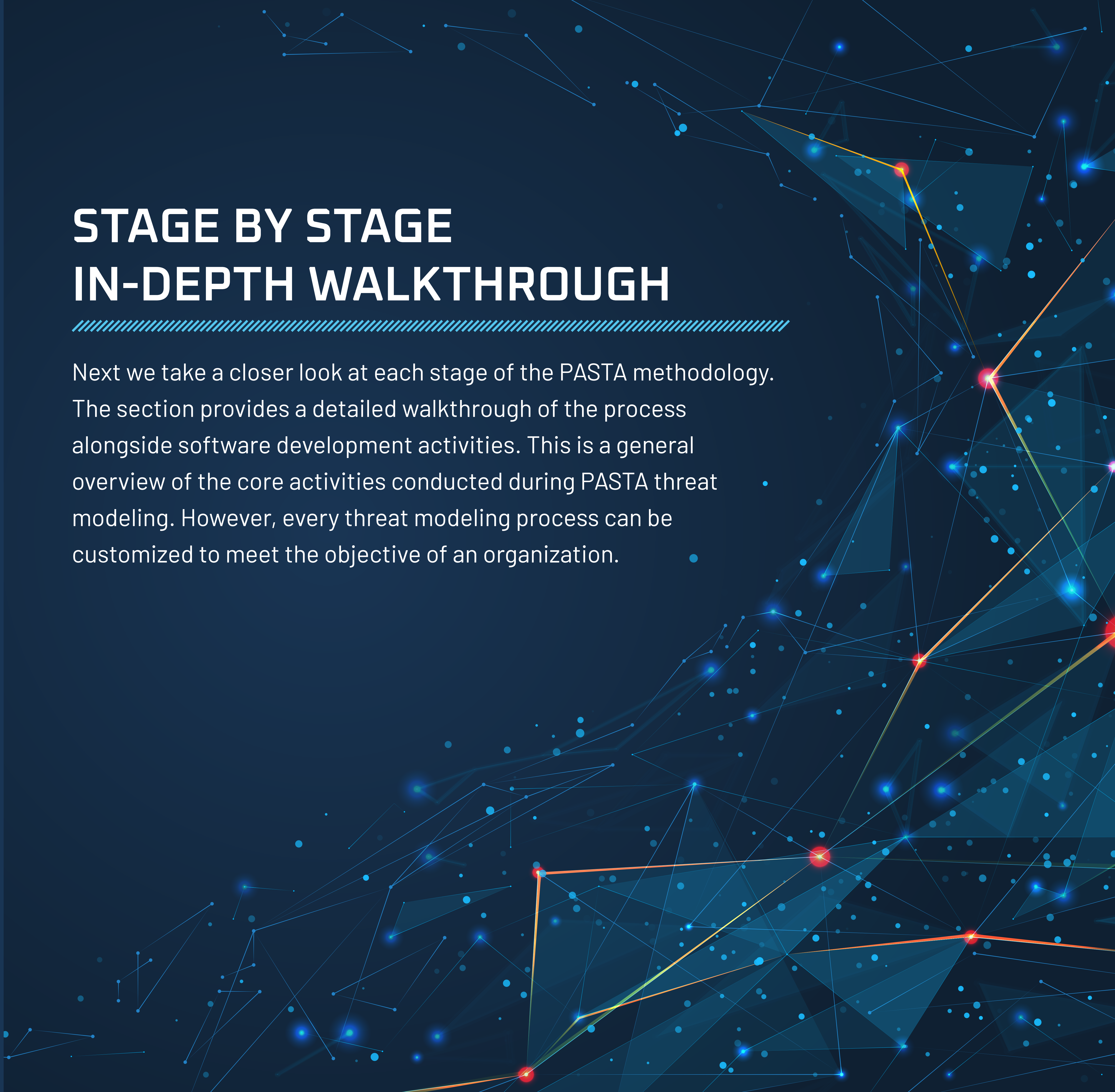
The goal of this stage is to provide mapped attacks and document how the vulnerabilities can be exploited by different attack methods.

STAGE VII: RESIDUAL RISK ANALYSIS & MANAGEMENT

The final stage of PASTA analyzes residual risks and business impact.

The goal of this stage is to quantify and qualify business impact, identify gaps in security controls, calculate residual risk, and provide risk mitigation strategies.

Organizations all over the world, like GitLab, are adopting PASTA as their internal threat modeling standard because of its risk-centric approach, collaborative tendencies, evidence-based threat intel, and focus on the probability of each attack.

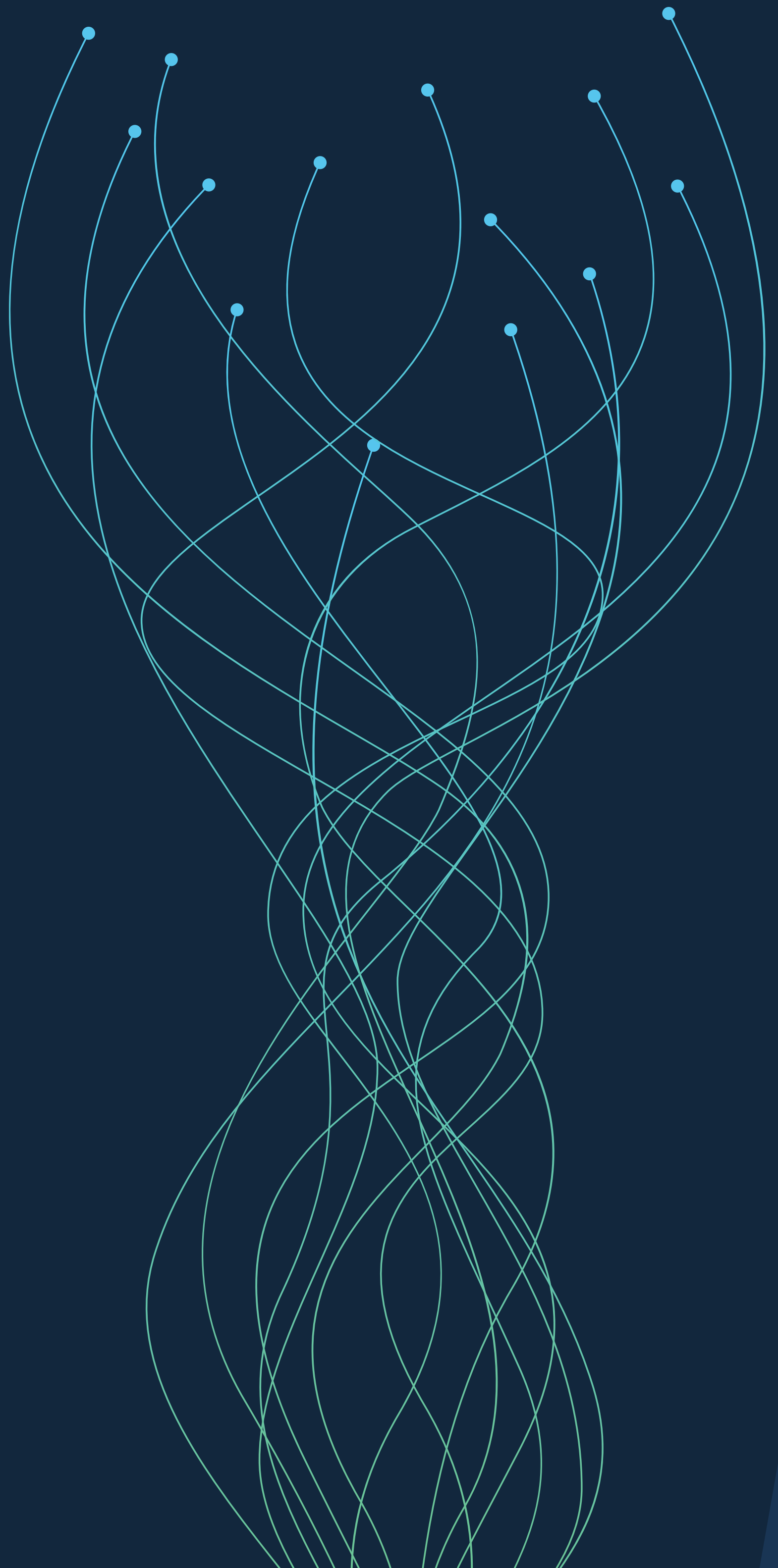


STAGE BY STAGE IN-DEPTH WALKTHROUGH

Next we take a closer look at each stage of the PASTA methodology. The section provides a detailed walkthrough of the process alongside software development activities. This is a general overview of the core activities conducted during PASTA threat modeling. However, every threat modeling process can be customized to meet the objective of an organization.

STAGE I

DEFINING OBJECTIVES



Central to PASTA's application threat modeling process are understanding business, financial, and operational objectives that legitimize an application's existence. From a business perspective, discovering why an application was developed, who the intended users are, and how and why features were developed is very important. Essentially, all of these questions map to requirements in any type of SDLC methodology.

Stage I process allows business objectives to be understood and security-related generalizations to be made in order to drive governance efforts that should be followed.

THE KEY GOALS OF THIS STAGE ARE:

- Defining principal business objectives to the application.
- Understanding the impact of an application and functional features to business.
- Developing a risk profile for the application.

STAGE I MAIN ACTIVITIES

1. DEFINE BUSINESS REQUIREMENTS

Here, specific information gathering activities are conducted in order to ensure that forward facing PASTA activities have a relevant anchor of context from which to build the overall risk analysis. Obtaining business requirements is essential in building a contextually sound threat model.

2. DEFINE SECURITY AND COMPLIANCE REQUIREMENTS

Meeting business objectives successfully means integrating both security requirements and compliance for the application being developed. The absence of security in the customer facing applications or products may spell trouble for many businesses. Similarly, ignoring or faltering on regulatory requirements for information security may introduce loss of accreditation or fines. This activity provides an opportunity to address both regulatory compliance and security requirements for the application being developed.

3. DEFINE BUSINESS IMPACT

What are the key business goals for the application? How long can an application be down for? What are the existing known risks? Is the data used by application sensitive or regulated? What service level agreements are in place?

These are just some questions being asked during this activity of Stage I to assess probable business impact. This assessment centers on deriving business impact from adverse events that can prevent or limit the predefined business objectives. This step goes beyond the commonly used Business Impact Analysis (BIA) utilized by a lot of companies, to consider potential revenue or estimated costs of impact. Identifying it allows PASTA to have relevant context when addressing threats, vulnerabilities, and attacks against an application.

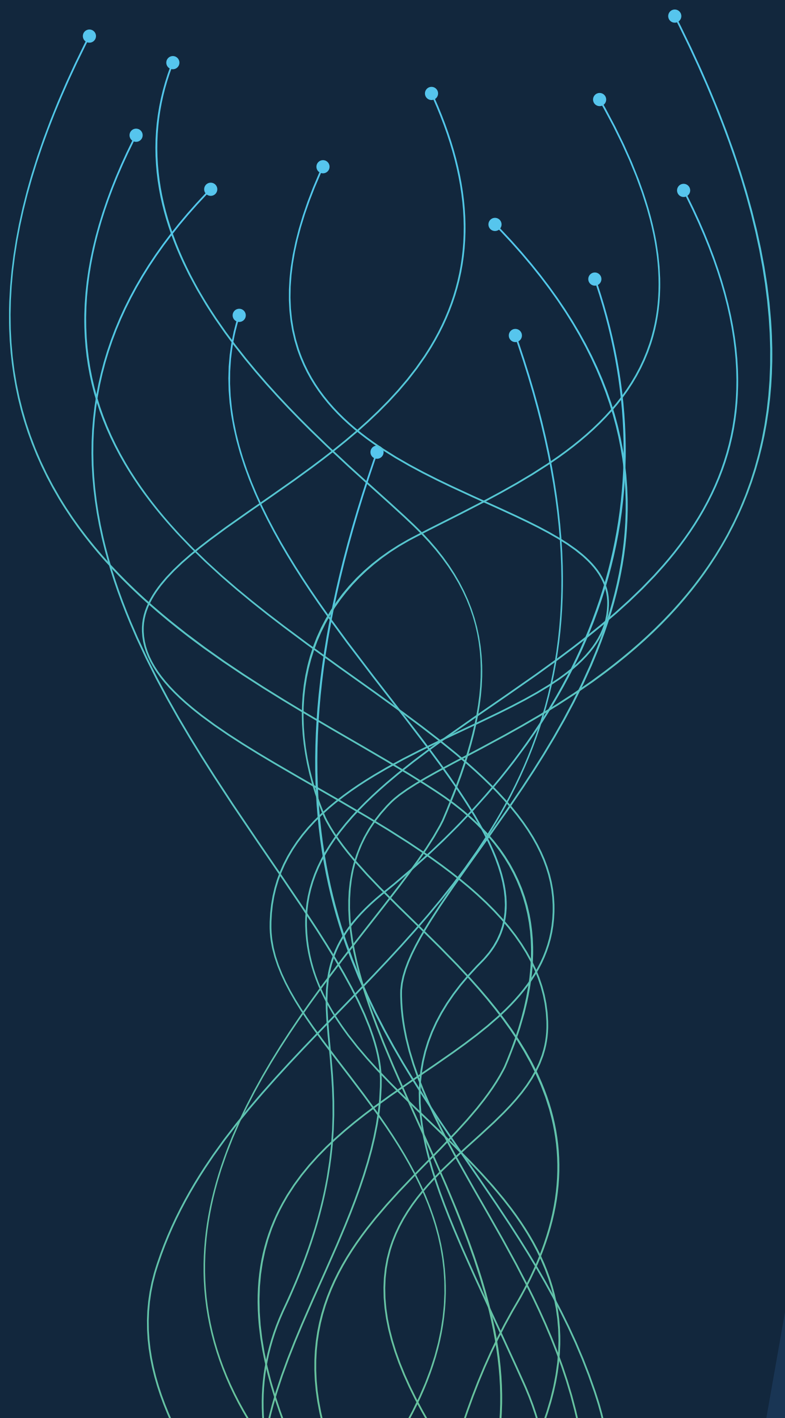
4. DEFINE RISK PROFILE

The goal of this activity is to identify and prepare a risk profile for the subject application. This exercise provides valuable insight to preexisting risk elements that have affected the application in the past. The risk profile seeks to comprehend what inherent risks affect the application from the vulnerability standpoint.

These are the four main activities in the Stage I of PASTA methodology. The output of the stage is developing of functional business and security requirements and awareness of business impact. It allows business objectives to be understood and security-related generalizations to be made in order to drive obvious governance efforts that should be followed.

STAGE II

DEFINING THE TECHNICAL SCOPE



In a risk-centric approach to application threat modeling, the focus is on protecting high-risk assets. PASTA helps identify whether the risk of an asset's compromise is worth more or less than the time and effort to develop countermeasures.

THE KEY GOALS OF THIS STAGE ARE:

- Identifying all of the assets in the application environment.
- Enumerating all types of hardware and software components that support use cases of the application.
- Building a baseline of security controls aimed at reducing the attack surface for each asset component contained within the scope of the threat model.

STAGE II MAIN ACTIVITIES

1. ENUMERATE SOFTWARE COMPONENTS

Providing a proper technical scope must begin with consideration of the data lifecycle within the application environment and the technology that ultimately sustains and secures it. This step looks at where the data originates, where it is stored, types of network segments it traverses, remote accesses, software interfacing with data, etc.

2. IDENTIFY ACTORS & DATA SINKS/SOURCES

This activity is focused on identifying smaller components that actually run or operate within the previously listed exercises. Here, we identify all databases, systems, and application actors who are making and receiving requests on each asset or across the application environment. Application architects and members of the development team should be able to create an initial list of actors and required services that sustain the application use contained within the environment. When searching for relevant data sources, these are best identified via application schematic design. In today's highly diverse world of data stores, understanding data flow and storage can be a treasure hunt.

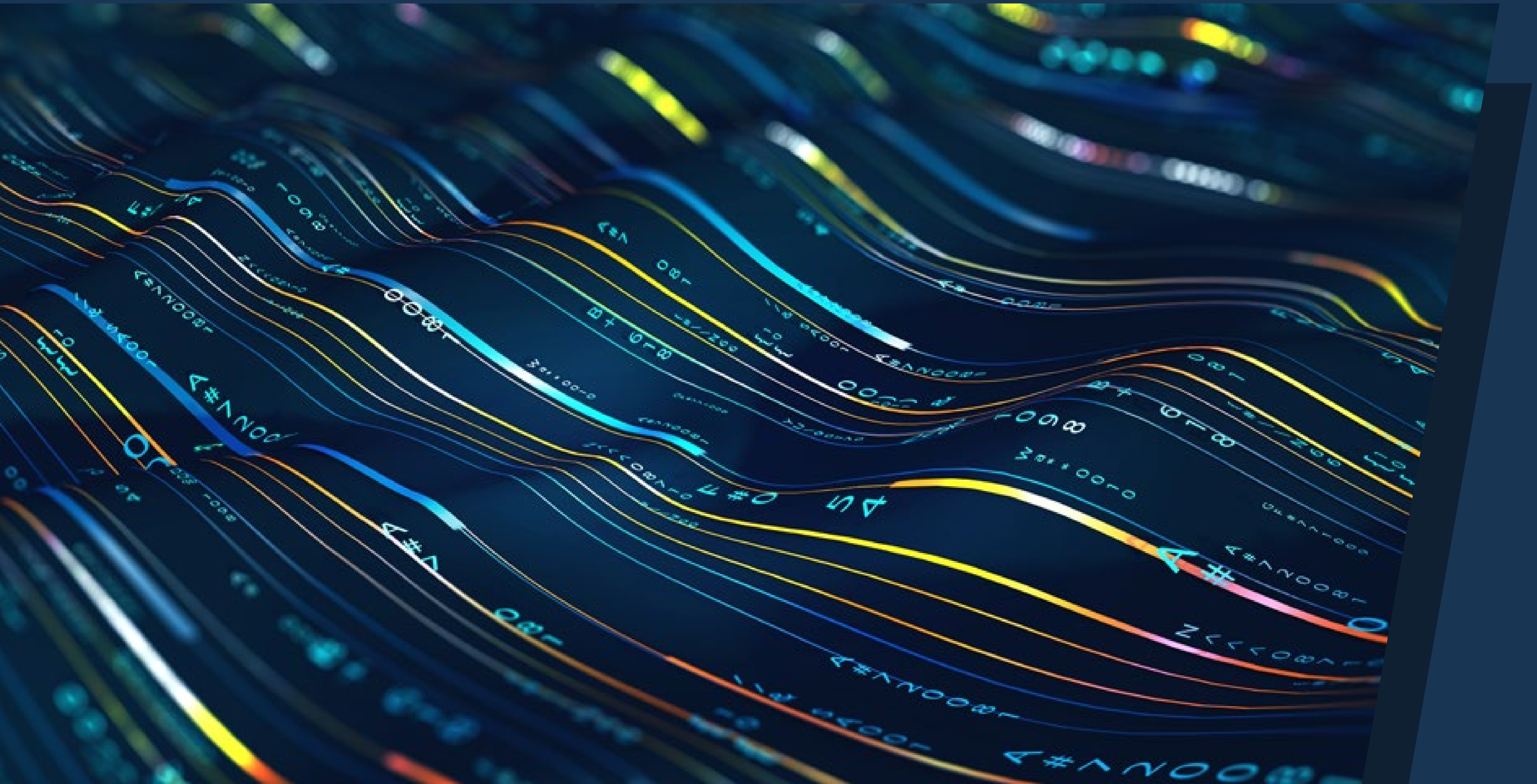
This stage utilizes mapping and e-discovery tools.

3. ENUMERATE SYSTEM-LEVEL SERVICES

Platform enumeration seeks to discover what system types or operating systems are being used. This may also include middleware. Service enumeration is aimed at identifying what services are running on the system assets on which much of the application will be running. These exercises require the use of some system commands and simple tools that facilitate identifying both services and actors in association with various with various use cases of the product application.

4. ENUMERATE THIRD-PARTY INFRASTRUCTURES

Since many application environments use various external networks, such as cloud-based PaaS, SaaS, IaaS, or third-party ASP and colocation models, it is important to bring third-party infrastructures into the technical scope. Here, we interview the third-party technology SMEs, review any existing network design documentation, and run quick tools to enumerate software used, data repositories, fingerprint platforms and services.



5. ASSERT COMPLETENESS OF SECURE TECHNICAL DESIGN

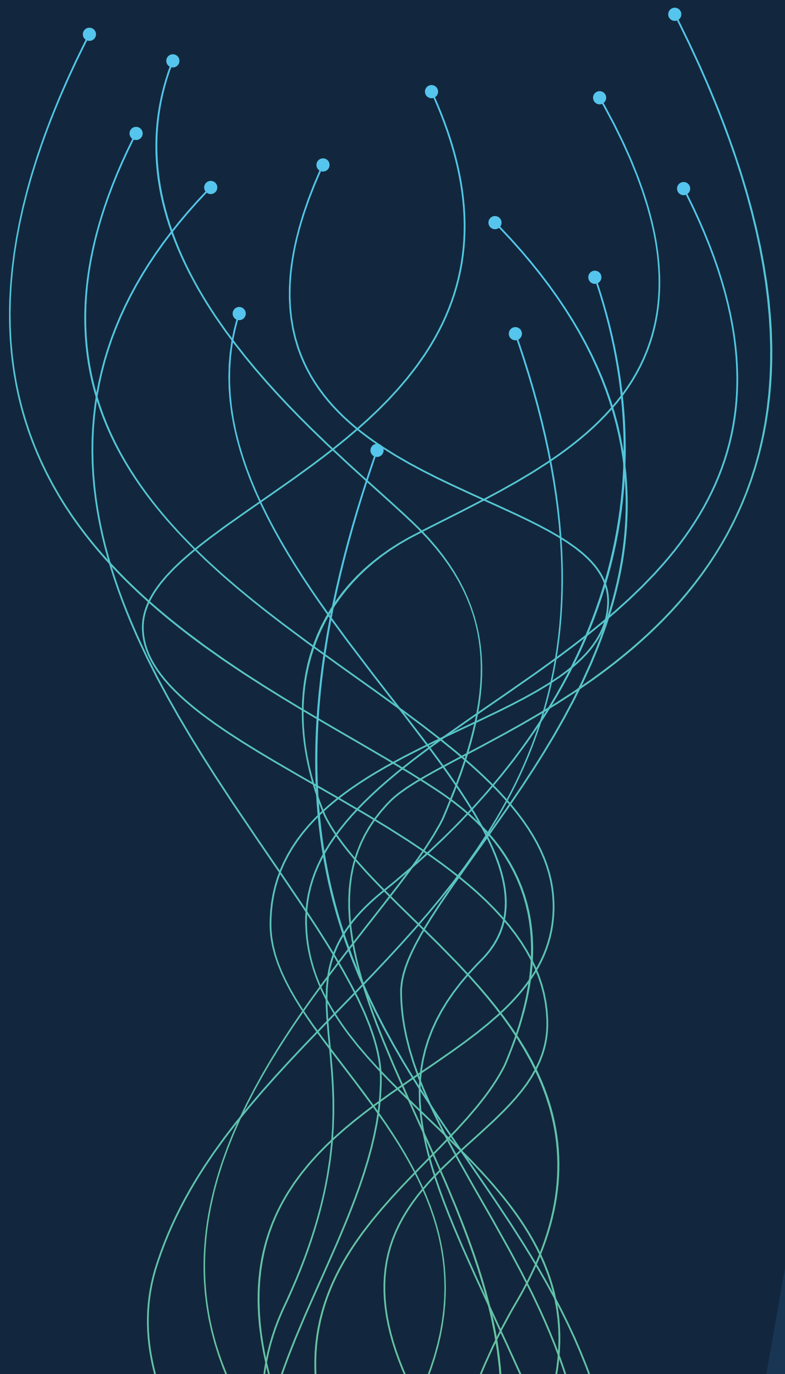
This activity aims to organize the identified application components and provides a level of security assurance that inherent risk mitigation strategies are applied to identified components from this stage. This activity efforts extend beyond enumeration and into applying some level of countermeasures that come from existing requirements for security.

It will also allow, further down in the process, for correlating vulnerability research or threat intelligence to the components that have been identified.

In conclusion, STAGE II **achieves** understanding of underlying technologies and related dependencies, **identifies** any applicable technology and security standards to be applied, and properly **defines** an application scope for the threat model to focus on. It helps determine potential exploits of technical vulnerabilities by the attack vectors identified as part of the attack analysis.

STAGE III

APPLICATION DECOMPOSITION & ANALYSIS



Application decomposition and analysis aims at correlating requirements and use case to other areas, including but not limited to network routing, hardware, configuration of systems and software, and database tuning. This stage is best conducted during the design stage, in which use cases become more clearly defined and connected to the underlying technology.

Application decomposition helps to ensure greater familiarity with how the use cases are designed to behave and related to other components of the threat model.

The correlation and analysis provide an opportunity for security threats to be better understood and properly mitigated in the subsequent PASTA stages.

STAGE III MAIN ACTIVITIES

1. ENUMERATE ALL APPLICATION USE CASES

Use cases enumeration visualizes the functional requirements. This benefits everyone involved – architects, developers, system engineers, business analysts, etc. The use cases reveal the actors and processes that are a part of the request. In turn, it provides the context for what the use cases are most critical to protect and later, allows for vulnerabilities and attack patterns to be appropriately mapped to them.

2. PERFORM DATA FLOW DIAGRAM EXERCISE OF IDENTIFIED COMPONENTS

Stage III of PASTA has the opportunity to ensure that each dissected area of the application supports the objectives defined in Stage I and adequately runs on the assets defined in the Stage II.

In this activity we build a DFD protocols and taxonomy of terms, which would be understandable by your team and one that can be leveraged to denote where abuse cases may take place due to excessive trust or insufficient number of countermeasures. Another point to keep in mind is that only certain aspects of application need to be DFDed. To be comprehensive, DFD should consider the following criteria: architectural (focuses on what activities are taking place across the application environment and how threat modeling components are interacting with one another), physical, and logical.

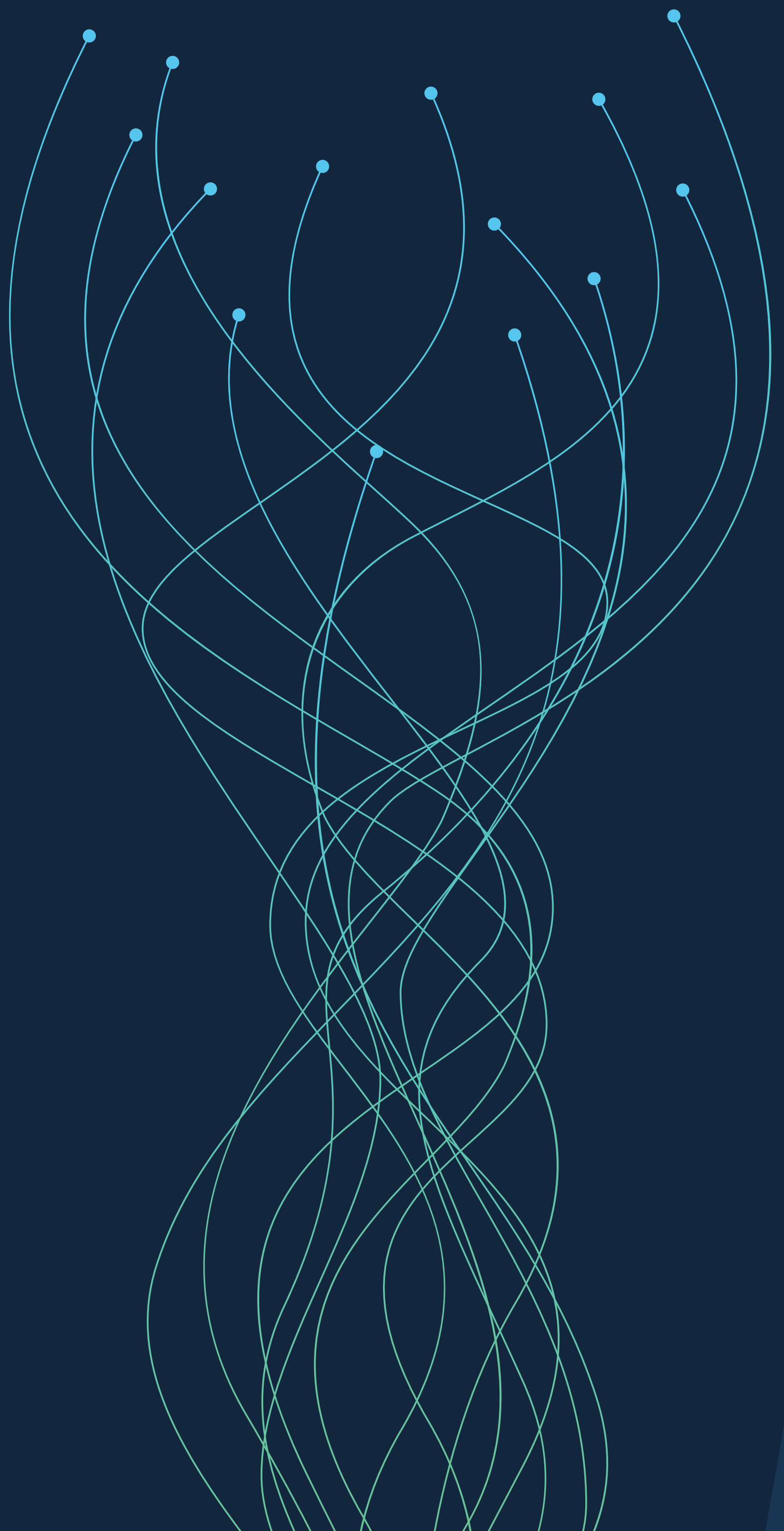
3. SECURITY FUNCTIONAL ANALYSIS AND THE USE OF TRUST BOUNDARIES

Building off the prior activities conducted thus far, this step allows to effectively build a "trust model" into the DFDs given what is known of application components, such as use, physical location, architectural positioning, inherent risk, criticality to the overall application environment, and presumed range of actors using each component. Trust boundaries introduce where new security countermeasures need to be developed, help all involved RACI (responsible, accountable, consulted, informed) participants see where and why distrust should be present within the application, and from which container threats can originate from.

In conclusion, decomposition is an essential step in every threat modeling methodology and provides the threat analyst with an understanding of the application from both the attacker and the defender perspectives. Here, PASTA provides considerations on where abuse cases may give way to data-focused attacks, platform continuity, authentication bypass, data integrity violations, and much more. The application decomposition is supported by the development of essential to the threat modeling documentation – Data Flow Diagramming and Trust Boundaries.

STAGE IV

THREAT ANALYSIS



What could be intended threats around the application? Is it really the data? Is it something greater? Could other aspects of the application environment really be the focus (human life, service downtime, etc.)? These are the key questions considered in the Threat Analysis stage. The stage focuses on the three major threat targets, which threat modeling can help protect: Data (PHI, PII, system information), Infrastructure (distributed attacks, persistence, etc.), and Human (wearable medical devices, wireless technologies, embedded software place human life within the realm of exploitation).

STAGE IV OBJECTIVES

- Review credible diverse source of threat data.
- Leverage internal sources of data, originating from security incidents, log and alert data.
- Enumerate likely threat agents who may be able to carry out supporting attack patterns for given threat.
- Identify the most likely threats to the application.
- Determine a threat likelihood value for each threat that is developed.

STAGE IV MAIN ACTIVITIES

1. ANALYZE THE OVERALL THREAT SCENARIO

The goal of this activity is to list threats against the application. Some of the threats are inherent to any type of a deployment model, while some threat patterns don't apply based on the application type and industry. The focus is on data, human, physical, and third-party targets within the application along with their associated threat motives.

2. GATHER THREAT INTELLIGENCE FROM INTERNAL SOURCES

Leveraging historical incident reports and security alert data. This provides a good basis for internal threat data to later develop into threat intelligence. The data can originate from internal log repositories or an Incident Response database.

3. GATHER THREAT INTELLIGENCE FROM EXTERNAL SOURCES

This activity aims to refine the list of possible threats against the subject application that is being threat modeled. External threat intelligence gathering is generally done by third-party managed security service providers (MSSPs).

4. UPDATE THE THREAT LIBRARIES

Attack libraries, such as MITRE's Common Attack Pattern Enumeration and Classification (CAPEC), are helpful to apply to applications and determine if any of the attack patterns are viable given various characteristics (known vulnerabilities, weaknesses, and threats). So, for this activity there are two main steps.

First, updating the threat library from source. CAPEC, OWASP, and Web Application Security Consortium (WASC) all have top threat listings to web and mobile applications.

Second, considering threat to attack relationship. Building off of the attack library updates and review, an attack tree at this stage begins to form and capture the threat to attack relationship.

5. THREAT AGENTS TO ASSETS MAPPING

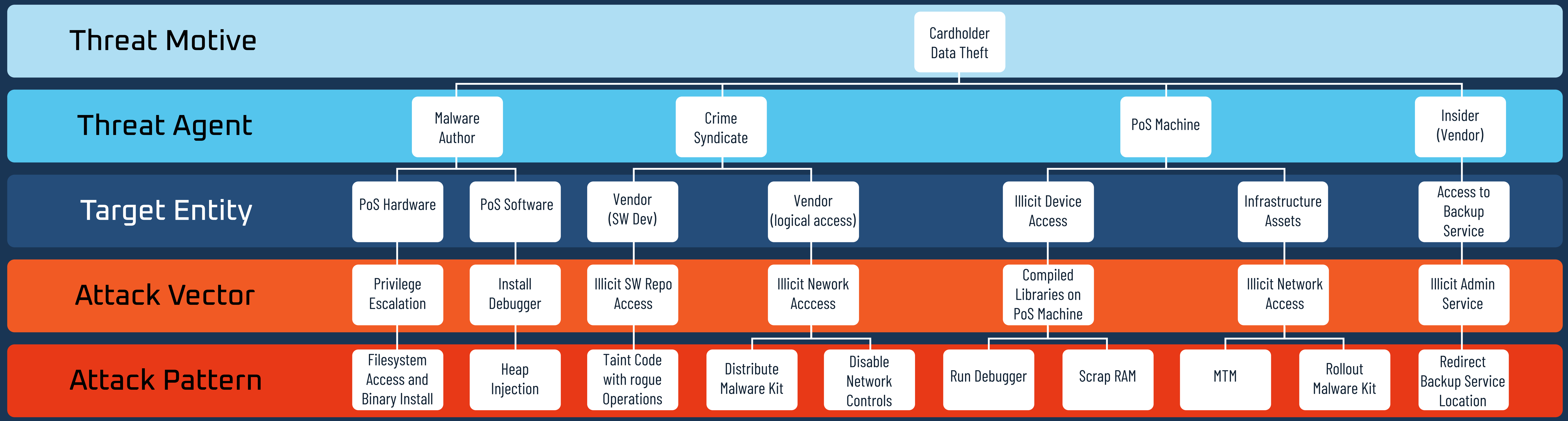
A threat agent is an individual or a group intending to launch a threat into action. Encompassing the threat agent into the threat model (mapping the agents to possible targets) is important because mitigation can apply to preventing or limiting the threat agent's actions through a range of countermeasures.

6. ASSIGN PROBABILISTIC VALUES AROUND THE IDENTIFIED THREATS

This activity helps assign a weighted percentage to each identified threat in the activity 1 of this stage. The probabilistic analysis is based on considerations for access, window of opportunity, ability to repudiate, risk reward (for threat agent), and threat simplicity. These five pieces of information represent whether or not some threats become actionable against a target application or application component.

Overall, threat intelligence will be a stage filled with research aimed at sustaining a threat model. The threat model will ultimately reflect a tree-like structure that has branches of assets, use cases, abuse cases, vulnerabilities, and attack patterns. These attack tree take full form later in the Stage VI. The key to this stage is collecting and analyzing good intelligence data from both internal and external sources.

RETAIL POS ATTACK TREE



SAMPLE THREAT POSSIBILITIES PER INDUSTRY

Utilities

Continuity Based Threats

- Disruption Bulk Energy Systems (BES) Software
- Infrastructure Denial of Service
- Insider Threat (Sabotage to BES)
- Malware Propagation

Telecommunications

Continuity & Confidentiality Based Threats

- Espionage or Spying
- Infrastructure Denial of Service
- Sources for Personally Identifiable Information (PII)
- Administrative Credentials
- Malware Propagation

Healthcare

Human Safety, Confidentiality, Availability

- Administrative Credentials
- Patient Portal Data
- Electronic Medical Records & Patient Health Information
- Access to Life Sustaining Systems
- Malware Propagation

Retail

Confidentiality, Integrity, Availability

- Cardholder Data
- Customer Personally Identifiable Information (PII)
- Financial Fraud (vendors, cost of goods sold, reporting)
- Downtime of System
- Malware Propagation

Gov't & Military

Confidentiality, Integrity, Availability, Human Safety

- Compromise Military Secrets
- Classified Intelligence Leaks
- Theft of Research & Development
- Malware Propagation

Hospitality

Confidentiality, Human Safety, Availability

- Targeted Attacks of Guests
- Blanketed Attacks of Guests/Passengers
- Guest/Passenger PII or Cardholder Data
- Availability of Systems
- Malware Propagation

Banking & Finance

Confidentiality, Integrity, Availability

- Account Information
- Customer Personally Identifiable Information (PII)
- Integrity of Financial Reporting
- Uptime of Financial Systems
- Malware Propagation

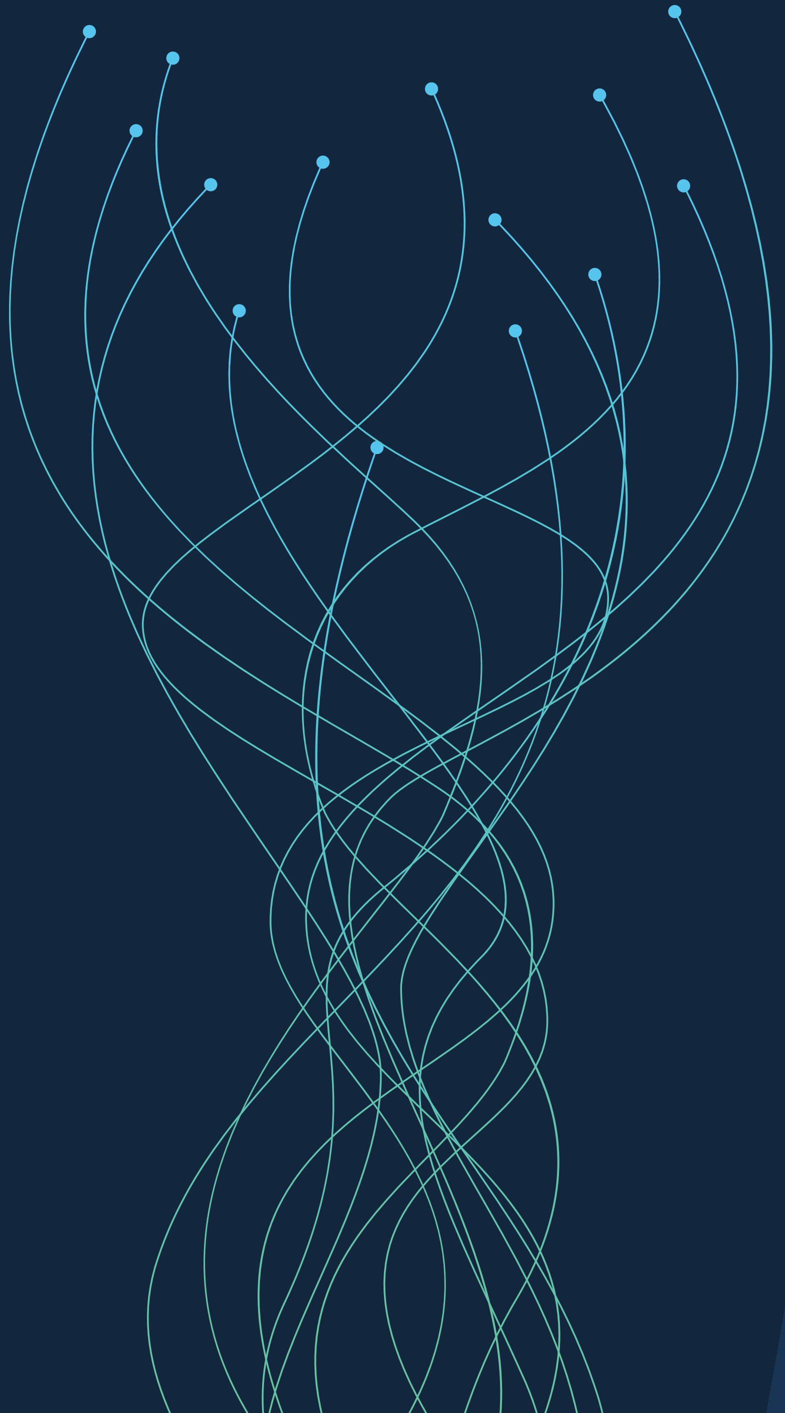
Software/Info Serv

Confidentiality, Integrity, Availability

- Source Code Leaks
- Source Code Sabotage
- Denial of Service
- Intellectual Property
- Malware Propagation

STAGE V

WEAKNESS AND VULNERABILITY ANALYSIS



Stage V of PASTA identifies vulnerable and weak areas across the application. The knowledge of inherent risks associated with the application, a refined technology scope, set of use cases, and relevant threats are now the background to discover what vulnerabilities and weaknesses maybe present across the application.

The **key objective** is mapping the information back to the attack tree that was introduced in the threat analysis activity of the previous stage. Vulnerability trimming is applied in order to port over relevant and confirmed vulnerabilities to the attack tree.

STAGE V MAIN ACTIVITIES

1. REVIEW AND CORRELATE EXISTING VULNERABILITY DATA

Establishing a historical context (no further than 12 months back) for what have been vulnerable or weak across application components is a key place to start Stage V. This activity provides a glimpse into the areas which have been historically more susceptible to exploitation. This information is used to align existing vulnerabilities and flows in the network or application design to the threat intelligence and data collected thus far.

Prior vulnerability data includes assets employed by the application, actors, software, services, third-party software, architecture, data sources, etc.

2. IDENTIFY WEAK DESIGN PATTERNS IN THE ARCHITECTURE

This activity revisits the DFDs developed in Stage III. The key architectural concerns around data security are examined in order to ensure that security is applied for data at rest, in transit, and while being processed. The key goal of this activity is security considerations being applied where trust boundaries have been formed.

3. MAP THREATS TO VULNERABILITIES

During this activity we start to map design flaws and software vulnerabilities to the branches on the attack tree. This "tree" is maintained as a visual representation of the relationship between vulnerabilities and threats. The mapping provides relationship nodes to the use cases affected by the threat. It, in turn, allows for further developing abuse case nodes, which map to use cases and plan out strategy for exploitation. Then, vulnerability nodes are added to provide a plausible point of entry or kink in the application armor and therefore supporting the abuse branch of the tree. This step produces a complete attack tree which is referenced in the following activities.

4. PROVIDE CONTEXTUAL RISK ANALYSIS BASED ON THREAT-VULNERABILITY

This activity helps to associate vulnerabilities to assets in the threat model and sustain the viability on how weaknesses and vulnerable components could facilitate the threats depicted. So, the goal of this step is to apply weaknesses and vulnerabilities to the attack tree for better context. This slowly builds prioritization model for remediation, based on relevance to use cases, asset components, and threat probabilities. The criterion around threat probabilities (introduced in Stage IV), combined with confirmed vulnerabilities can begin to demonstrate the viability of successful abuse cases.

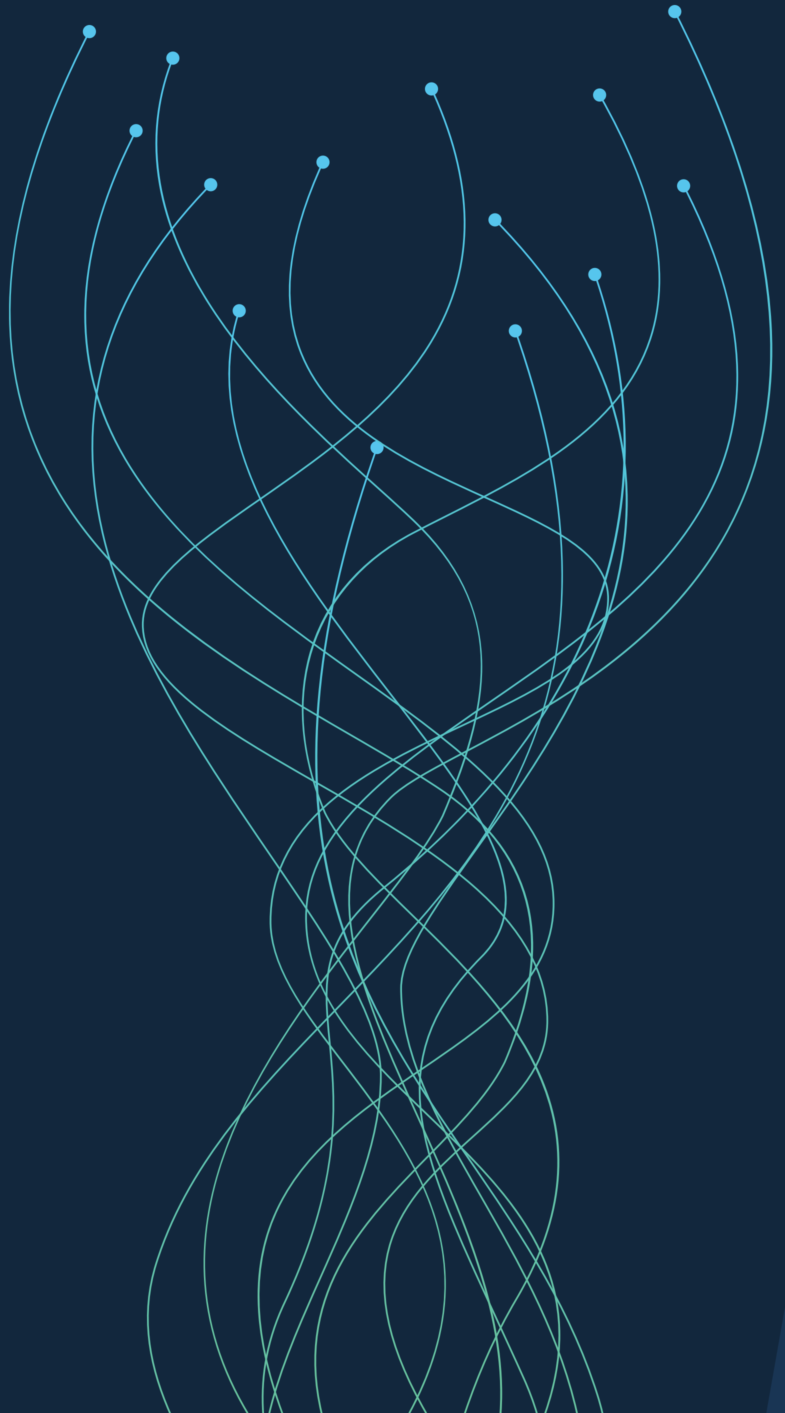
5. CONDUCT TARGETED VULNERABILITY TESTING

During this activity we conduct targeted vulnerability testing to evaluate components that are within the application scope and in support of the overall threat model. Vulnerabilities are selected based on threat relevance and application component. Various active and passive network and application scanners are used to conduct the vulnerability scans. Targeted vulnerability scanning makes the results much more topical to the threat model and evolving attack tree that is being built across the PASTA process.

To summarize, Stage V of the methodology allows to map vulnerabilities to threats, as well as determine risk severity of existing vulnerabilities and design flaws from architectural analysis. We are able to prioritize security test cases for specific vulnerabilities and flaws and conduct targeted and effective testing.

STAGE VI

ATTACK MODELING & SIMULATION



This stage's key objective is to complete the attack tree, which is a centerpiece for the application threat modeling. It organizes all of the information around threats, target assets, abuse cases, vulnerabilities, design flaws, and the exploits that profit from all of the above.

Being a risk-centric threat modeling approach, here we determine the probability for a vulnerability to be exploited. In the last stage, we identified relevant vulnerabilities that relate to both the desired threat as well as the targeted asset. In this stage, we model the different types of attacks that would exploit the relevant vulnerabilities for the application, all aimed at realizing the various threat objectives.

An attack tree is a key to threat modeling. A well-defined attack tree shows the scope of assets, actors, services, and other entities defined in the other stages that came before Threat Analysis (stage IV). Completed, it reveals different layers of attacks, all mapped to preexisting vulnerabilities that facilitate the exploitation data, credentials, and simply online reliability of the application or system.

STAGE VI MAIN ACTIVITIES

1. ANALYZE POSSIBLE ATTACK SCENARIOS

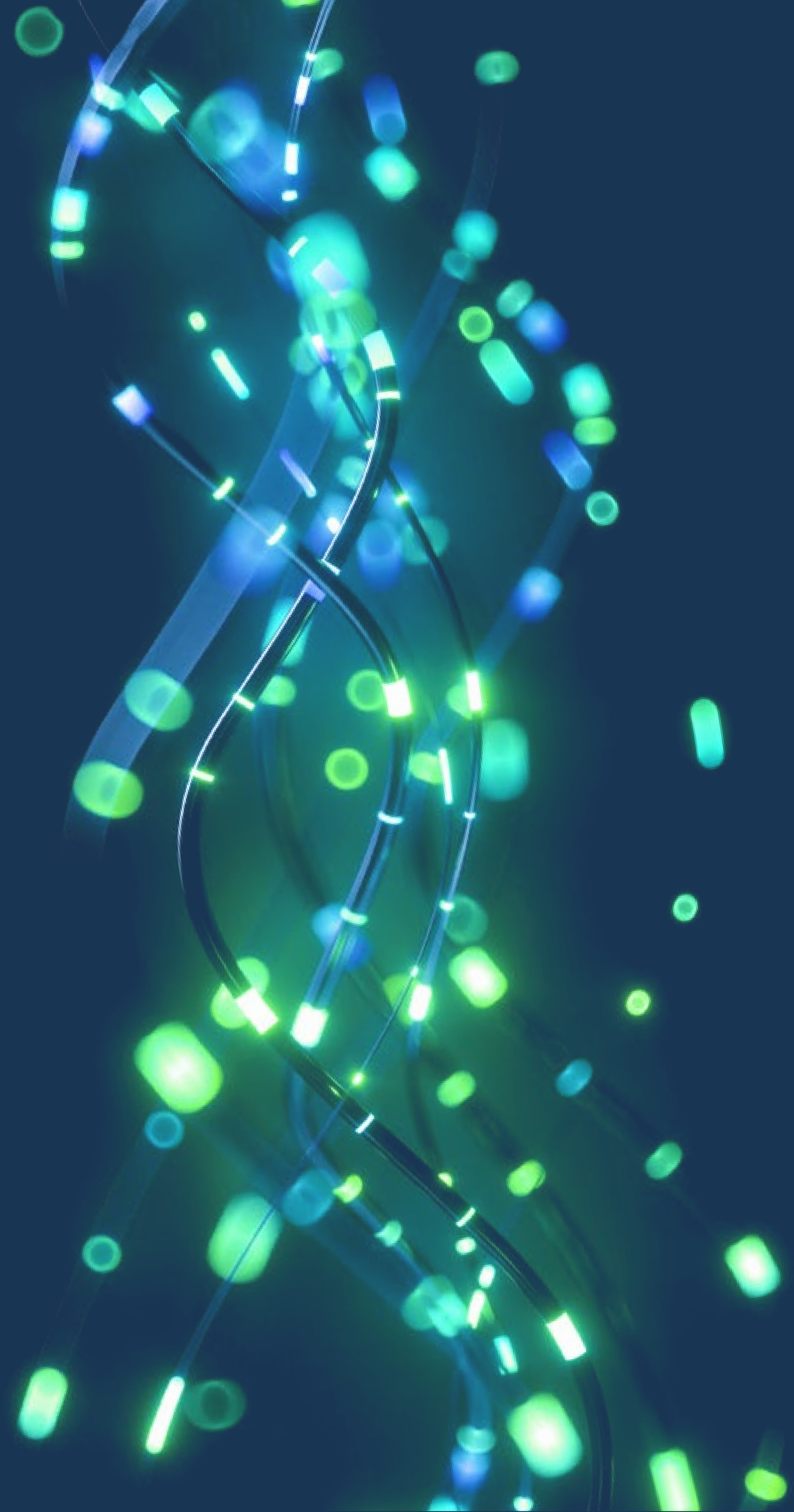
This activity enumerates and analyzes possible attack scenarios, while considering the target assets and related threat components. It builds off of the selected threats that have been substantiated in stage IV of PASTA methodology. The end goal is to develop a list of attack scenarios that are operationally possible and technically feasible given the known vulnerabilities in the environment.

2. UPDATE THE ATTACK LIBRARY/VECTORS AND THE CONTROL FRAMEWORK

The focus of this activity is on ensuring that the list of attacks as well as possible control measures is vast enough in order to properly build a threat model – a key objective for this stage. Here, we search for a comprehensive library of attack patterns and develop a list of possible controls as a part of a broader control framework.

3. IDENTIFY THE ATTACK SURFACE AND ENUMERATE THE ATTACK VECTORS

The goal of this step is to identify the full attack surface for the scope of the threat model using the visualization of the attack trees, which provide a



visual representation of the relationships among attacks and their vulnerability counterparts. The attack surface encompasses each of the possible attacks that can exploit vulnerabilities identified in the previous stages. This activity ensures that the attack tree is finalized.

4. ASSESS THE PROBABILITY AND IMPACT OF EACH ATTACK SCENARIO

In this activity, we review the attack scenarios and determine the probability for success around exploitation. This pseudo-probabilistic analysis is useful in identifying where the most urgent part of threat mitigation should take place. Determining the probability depends on attack prerequisites, weakness and vulnerability maturity, hackability, severity rating. The prior mentioned CAPEC library is employed in this step to provide information on the severity and the difficulty associated with exploiting a target vulnerability.

Another key objective under this activity is to determine impact. The impact is tied to the assumption that the attack pattern is successful. We must calculate what the adverse effects are to the business objectives for the application, as well as the specific use cases affected by the attack pattern.

5. DERIVE A SET OF CASES TO TEST EXISTING COUNTERMEASURES

Now that the attack tree is completed, it is time to check of the control framework in place can provide some level of inherent threat mitigation. Here, we select any possible controls that could limit or eliminate any aspect of the threat. This includes the opportunity for an abuse case, the existence of a vulnerability or a design weakness, or the attack itself.

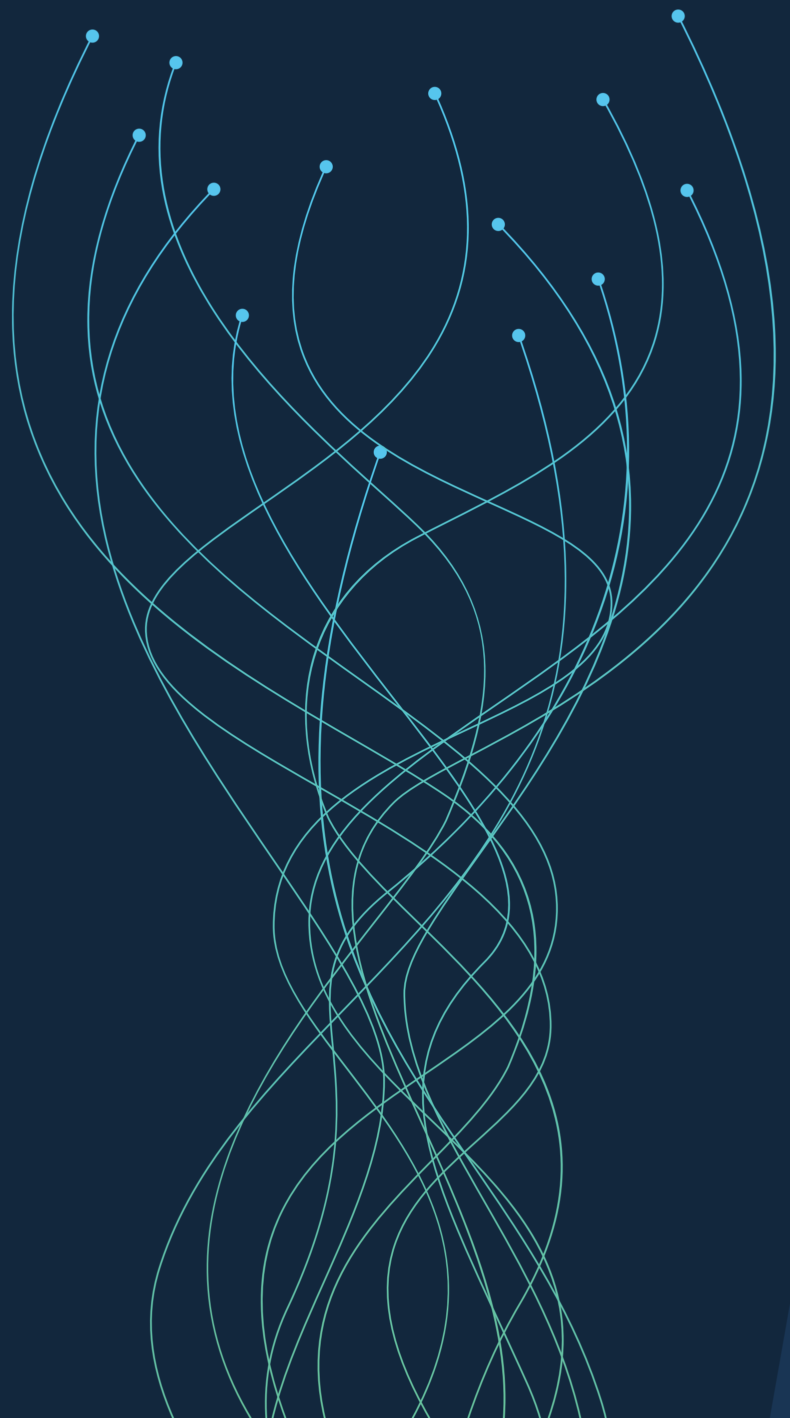
6. CONDUCT ATTACK DRIVEN SECURITY TESTS AND SIMULATIONS

The key objective for this activity is to demonstrate attack viability by denoting the probability and severity level of the attacks defined in the attack tree. This activity helps illustrate which attack pattern may be successful. The values, derived through the testing and simulations, are used as a part of the overall residual risk formula that is introduced and calculated in Stage VII. The threat modelers define the scope of the penetration tests, which focus only on the vulnerabilities identified in the prior stage.

STAGE VI of the PASTA methodology achieves the complete attack tree, which provides a kill-chain model of the attack scenarios. It determines use and abuse case and updates the attack libraries and vectors. A series of targeted attack testing and attack simulations is performed in this stage.

STAGE VII

RESIDUAL RISK ANALYSIS AND MANAGEMENT



Stage VII of the PASTA is focused on mitigating threats that matter to the application, product team, and overall business. This is achieved by applying all types of countermeasures that are both effective and topical to the threats and attacks depicted under the PASTA threat model.

BENEFITS OF THIS APPROACH

- Time saving.
- Evidence based.
- Fostering of a greater understanding of how security impacts application use cases.
- Understanding of impacts to business objectives.
- Development of a mature security framework.

STAGE VII MAIN ACTIVITIES

1. CALCULATE THE RISK OF EACH THREAT

During this activity, the risk professionals review the threat model and the supporting attack trees in order to see how viable the identified threat patterns are. For each threat in the attack trees that are depicted, a percentage weight of probability should be assigned based on internal threat data, external threat intelligence, and viability of attacks.

2. IDENTIFY THE COUNTERMEASURES

The objective of this activity is to determine the right amount of risk mitigation through the use of agreed upon countermeasures to be developed or designed. Software engineers and architects must work together to establish to how unacceptable risk level are to be addressed by the implementation of newly proposed countermeasures.

This activity revisits the threat model and the associated attack tree on order to see what kind of countermeasures can be implemented through the architectural design or software updates to either the codebase or product application.

3. CALCULATE THE RESIDUAL RISKS

The objective of this activity revolves around risk analysis. The asset-centric approach addresses risk through variables, which are not traditionally covered in risk analysis: probabilities (coefficient used to determine the likelihood of the vulnerabilities exploited under the observed conditions) and countermeasures (measures that provide some degree of protection against the threat). Considering these variables, the risk-centric threat modeling defines the residual risk.

4. RECOMMEND STRATEGY TO MANAGE RISKS

This activity is used to update risk profiles associated with the system or application in order to have a current account of both risk and risk strategy. Compliance and risk management teams should work together with the threat modeling efforts to update the risk profile of the application and capture the risk values. These risk values can be factored into any formal enterprise risk management suite for tracking and reporting.

Stage VII of the PASTA methodology provides cost-effective countermeasures, based on residual risks analysis for each threat while considering technical and business impacts. The stage produces a list of countermeasures and recommended risk mitigation options. Risk mitigation strategies for each threat scenario are recommended.

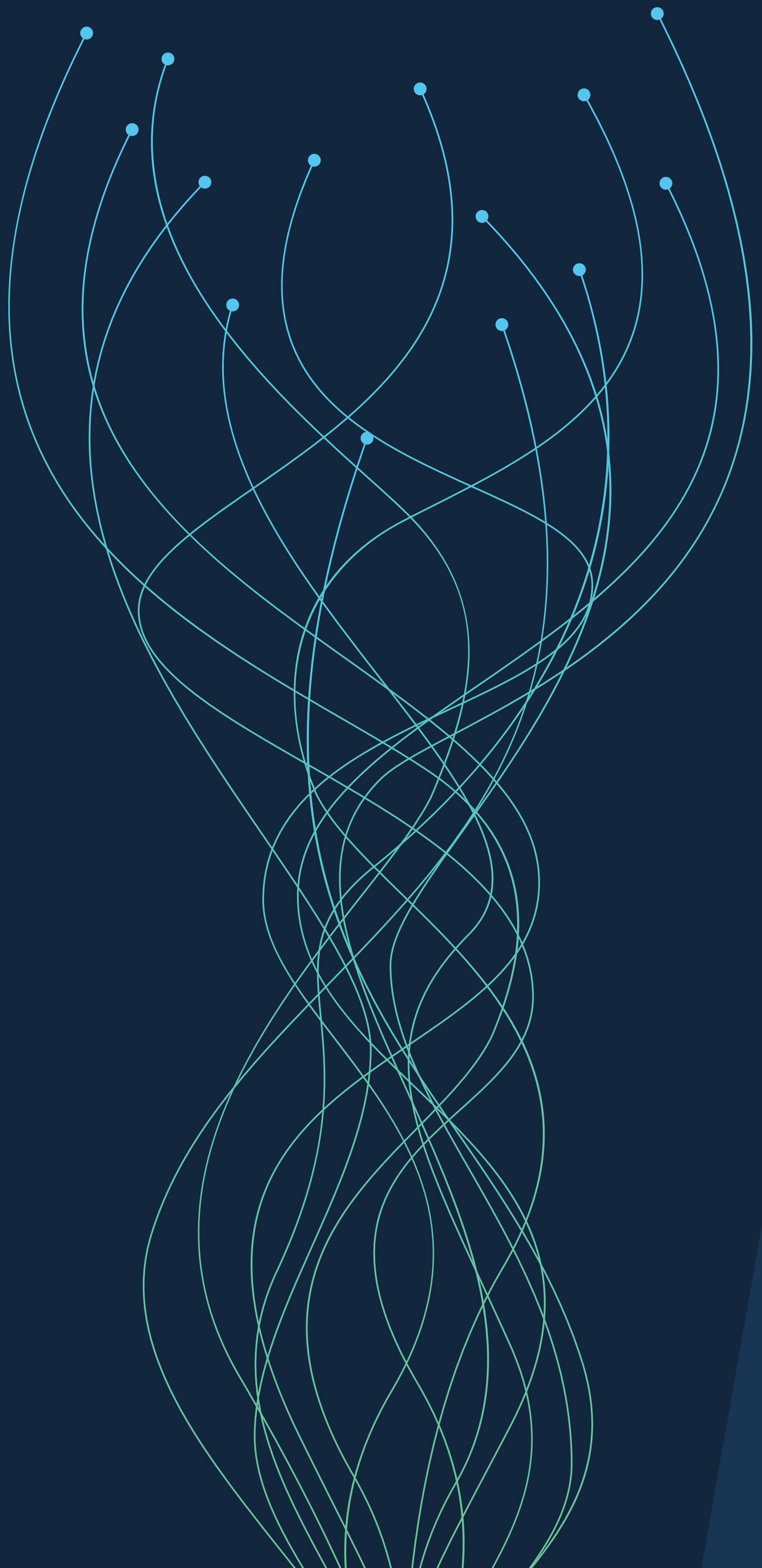
PASTA METHODOLOGY SUMMARY

Regardless of the organizational size, application threat modeling through PASTA methodology allows for development of a clear roadmap for risk mitigation – a message that is understood and appreciated by all stakeholders.

PASTA provides the framework for integrating existing information security, security engineering, and risk management disciplines. PASTA's uniqueness stands on the holistic view of threats and attacks to the application (as well as organizational) environment, consideration of the business aspects of mitigating risks posed by cyber threats, and developing security solutions to fit application and business objectives. The methodology fills the current gap between technical and business risk analysis disciplines when addressing cyber threats. It saves organizations time and mitigation cost by addressing security issues in the SDLC and offering actionable solutions and improvements to risk management frameworks that correlate with application objectives and business goals in general.

PASTA allows corporations to evolve vulnerability assessments of threats and attack analysis to the drivers for determining the risk mitigation strategy.

WHY ADOPT PASTA THREAT MODELING?



Threat models are often used by security champions to discover flaws in application environments, systematically exploiting applications (mobile, IoT, etc.) for security purposes. Threat modeling allows to acknowledge security concerns and create appropriate countermeasures before they are exploited by the threat actors. PASTA is a risk-centric threat modeling methodology that provides a step-by-step process to inject risk analysis and context into an organization's overall security strategy from the beginning. It encourages collaboration across all stakeholders, creating an environment focused on security.

The hardest activity to perform for an organization is often risk prioritization. PASTA provides security teams with prescriptive guidance on where to focus mitigation efforts. It proves the viability of impact to developers in stage six and demonstrates the business impact that is inherent in stage one, that affects their technology aspects as defined in stage two, then it allows to analyze the residual risks.

This approach to risk-based threat modeling provides developers, executives, and board members with clear visibility into the risks, flaws, and vulnerabilities of your application. PASTA can be the backbone of a security program, woven into every layer of an organization and application development process and allows a company to integrate security into its culture.

VERSPRITE

Founded in 2007 and headquartered in Atlanta, Georgia, VerSprite is a global leader in risk based cybersecurity consulting services and PASTA threat modeling. We enable businesses to improve protection of critical assets, ensuring compliance and managing risk. With operations and clients that span 15 countries and 4 continents, our squad of security experts quietly protect and guide organizations to adapt before threats become incidents. Helping organizations navigate the complex cyberthreat landscape with customizable and affordable cybersecurity solutions. At VerSprite, we believe in proactive defense through offensive security.

BEGIN YOUR JOURNEY

sales@versprite.com | versprite.com