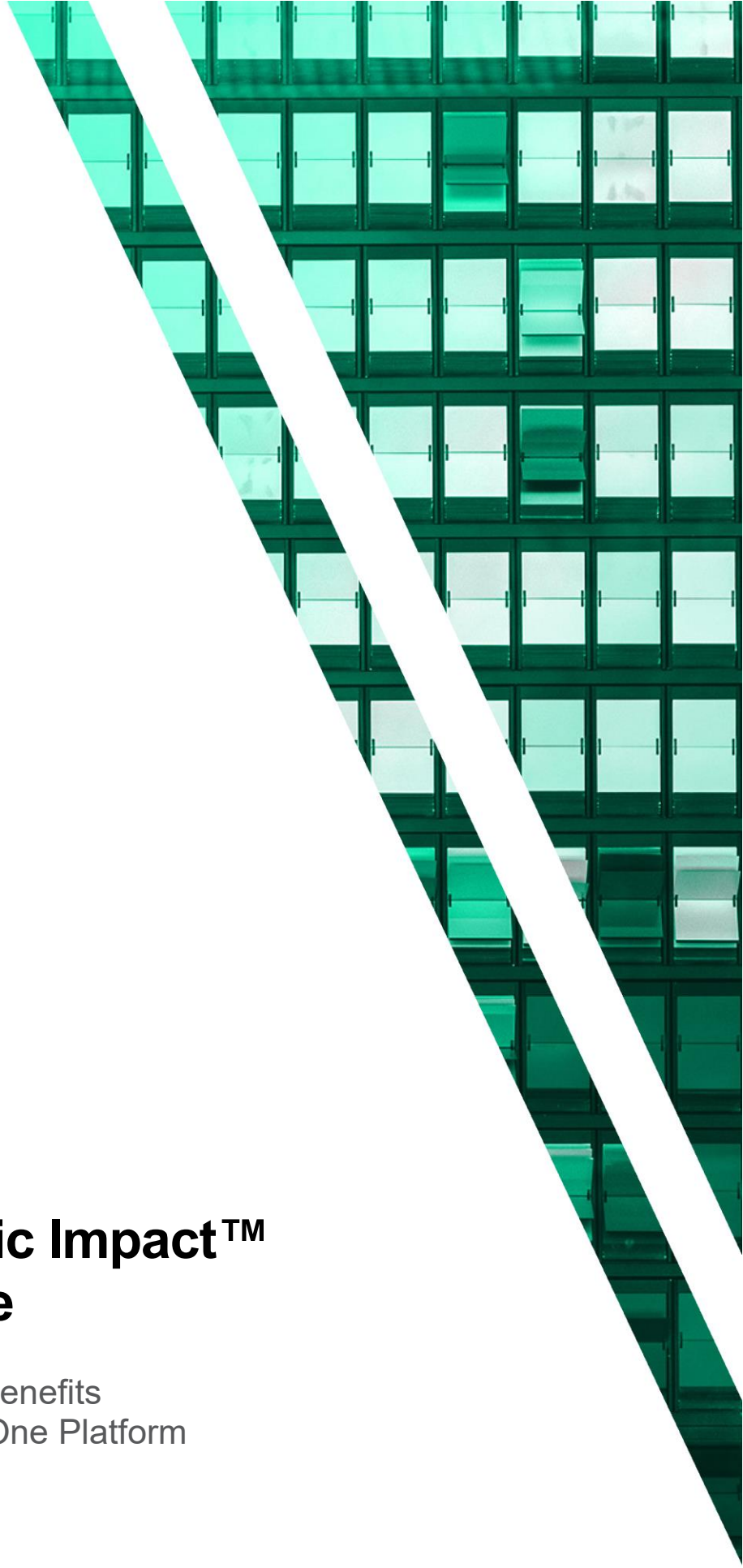
FORRESTER®



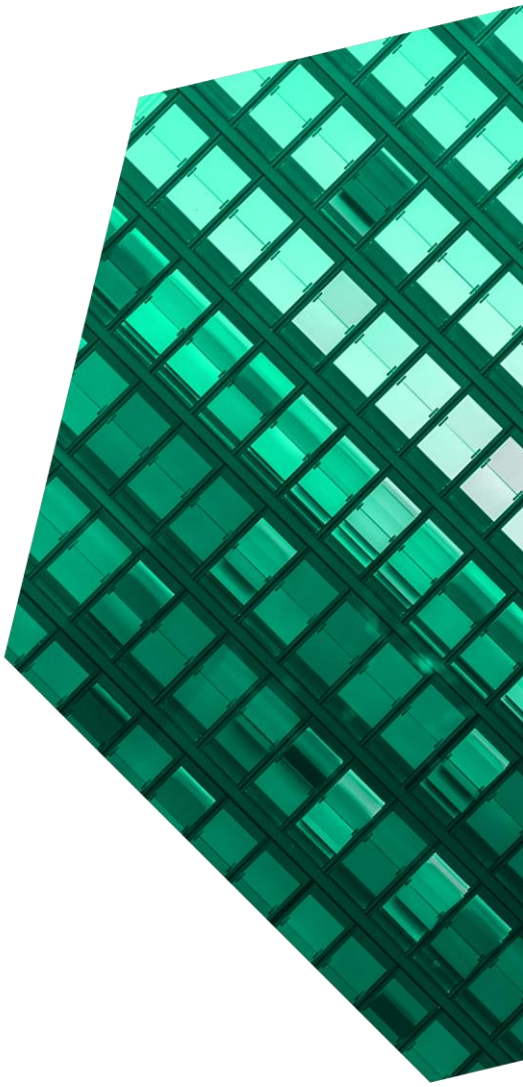
The Total Economic Impact™ Of Checkmarx One

Cost Savings And Business Benefits Enabled By The Checkmarx One Platform

MARCH 2024

Table Of Contents

Consulting Team: Luca Son
Kara Luk
Marianne Friis



ABOUT FORRESTER CONSULTING

Forrester provides independent and objective research-based consulting to help leaders deliver key outcomes. Fueled by our customer-obsessed research, Forrester's seasoned consultants partner with leaders to execute their specific priorities using a unique engagement model that ensures lasting impact. For more information, visit forrester.com/consulting.

© Forrester Research, Inc. All rights reserved. Unauthorized reproduction is strictly prohibited. Information is based on best available resources. Opinions reflect judgment at the time and are subject to change. Forrester®, Technographics®, Forrester Wave, and Total Economic Impact are trademarks of Forrester Research, Inc. All other trademarks are the property of their respective companies.

Executive Summary

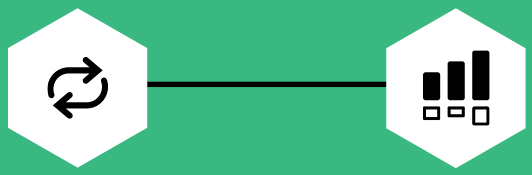
All companies rely on software to power their business, connect with customers and partners, automate back-office processes, and extend market presence. Developers tasked with coding such business-critical software face immense pressure to provide value to customers faster than ever before.[1] Checkmarx One helps organizations embed application security into the software development lifecycle to enhance security and improve time to value for their applications.

Checkmarx One is an application security platform that helps organizations streamline the process of developing secure code and applications. Security and development teams use Checkmarx One to detect and manage vulnerabilities at every stage of the software development lifecycle. Checkmarx One integrates a comprehensive suite of AppSec solutions, including static application security testing (SAST), software composition analysis (SCA), software supply chain security (SSCS), API security, dynamic application security testing (DAST), container security, and infrastructure-as-code (IaC) security.

Checkmarx commissioned Forrester Consulting to conduct a Total Economic Impact™ (TEI) study and examine the potential return on investment (ROI) enterprises may realize by deploying Checkmarx.[2] The purpose of this study is to provide readers with a framework to evaluate the potential financial impact of Checkmarx on their organizations.

To better understand the benefits, costs, and risks associated with this investment, Forrester interviewed eight representatives with experience using Checkmarx. For the purposes of this study, Forrester aggregated the interviewees' experiences and combined the results into a single composite organization that is a global company with $10 billion in annual revenue. The composite has 1,000 developers who develop and maintain 1,000 applications annually.

KEY STATISTICS



Return on investment (ROI)
177%

Net present value (NPV)
$4.55M

Prior to using Checkmarx, the interviewees noted how their organizations used a mixture of legacy application security tools and relied on manual code scanning across application teams. However, prior environments yielded limited success, leaving them with manual scanning limitations, high false-positive rates, difficulty understanding vulnerability impact and root causes, and a lack of support for necessary coding languages, frameworks, and libraries. These limitations made it difficult for developers and security teams to efficiently identify and remediate vulnerabilities, reducing risk posture, operational speed, and scalability.

After the investment in Checkmarx, the interviewees' organizations shifted left by embedding vulnerability detection capabilities earlier into the software development lifecycle. Key results from the investment include improved efficiency for developers and application security analysts, reduced risk of data breaches, and technology cost consolidation.

KEY FINDINGS

Quantified benefits. Three-year, risk-adjusted present value (PV) quantified benefits for the composite organization include:

- **Developer productivity improvement of 40% to 50% for security tasks.** With Checkmarx, the composite organization integrates automated code scanning into developer workflows and the software development lifecycle, and developers gain richer context for identifying vulnerabilities and remediation paths. This enables developers to avoid manual code review work, reduces investigation and remediation efforts, decreases rework, and improves coding security over time. For the composite, the productivity improvement results in $3.4 million in developer labor cost savings over three years.

Developer productivity improvement for security tasks

40% to 50%

- **Security analyst efficiency improvement of 30% to 40%.** Checkmarx enables application security teams at the composite organization to efficiently review code, collaborate with developers to address misconfigurations and security issues, and provide reporting for compliance and audit purposes. By improving secure coding practices, security teams also have fewer issues to address, saving additional effort. For the composite organization, the efficiency improvement results in $1.1 million in application security analyst labor savings over three years.
- **Risk of a data breach is reduced by 25% to 35%.** Checkmarx reduces the risk of a data breach for the composite organization year over year by improving risk detection and mitigation processes and improving secure coding practices. With automated code scanning and improved context around vulnerabilities, developers identify and address code issues earlier in the software development lifecycle. This enables the composite to harden applications over time and strengthen developer knowledge in secure coding practices, enhancing the overall security of the organization's applications and reducing risk of breach. For the composite, the risk reduction is worth $1.5 million in avoided breach costs over three years.
- **Technology cost consolidation saves $453,000 annually.** The composite organization retires legacy testing tools with the adoption of Checkmarx. Over three years, the organization avoids $1.1 million in licensing, IT maintenance, and support costs for these legacy solutions.

Unquantified benefits. Benefits that provide value for the composite organization but are not quantified in this study include:

- Improved developer velocity and time-to-value.
- Improved developer experience and developer training.
- Improved ability to meet compliance requirements.

Costs. Three-year, risk-adjusted PV costs for the composite organization include:

- **Licensing costs of $1.4 million over three years.** The composite organization pays $554,400 in annual licensing costs, based on the number of Checkmarx users.
- **Implementation, training, and ongoing management costs of $1.2 million over three years.** The composite organization incurs costs

for professional services and internal labor associated with the implementation, training, and ongoing maintenance.

The representative interviews and financial analysis found that a composite organization experiences benefits of $7.13 million over three years versus costs of $2.57 million, adding up to a net present value (NPV) of $4.55 million and an ROI of 177%.

“With the rise of cybercrime and the cost of data breaches today, the value of a tool like this is undeniable, and the tool needs to be in place. We need to be shifting security to be locked into our software development lifecycle so that we can improve our security posture and reduce the risk of being attacked and potentially breached.”

Application security engineer, manufacturing

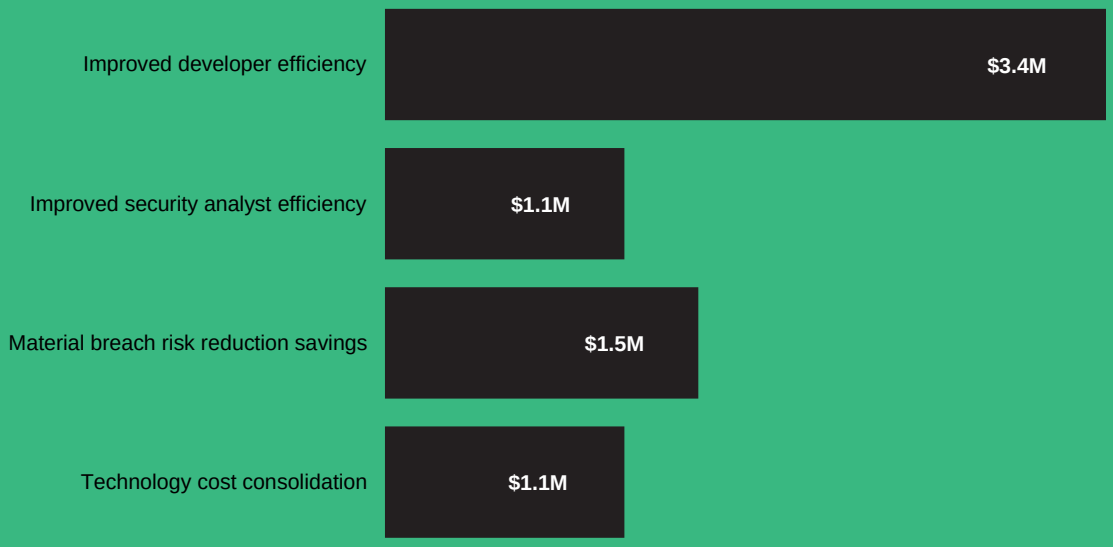
ROI
177%

BENEFITS PV
$7.13M

NPV
$4.55M

PAYBACK
<6 months

Benefits (Three-Year)



“[Checkmarx] empowers our developers to start understanding the importance of application security. It gives them every tool they need to be successful in that effort, and it overall is just a huge gain on our [security] front. It’s a seamless tool … and it makes sense why they’re a leader in the space.”

— Application security engineer, manufacturing

TEI FRAMEWORK AND METHODOLOGY

From the information provided in the interviews, Forrester constructed a Total Economic Impact™ framework for those organizations considering an investment in Checkmarx.

The objective of the framework is to identify the cost, benefit, flexibility, and risk factors that affect the investment decision. Forrester took a multistep approach to evaluate the impact that Checkmarx can have on an organization.

DUE DILIGENCE

Interviewed Checkmarx stakeholders and Forrester analysts to gather data relative to Checkmarx.

INTERVIEWS

Interviewed eight organizations using Checkmarx to obtain data with respect to costs, benefits, and risks.

COMPOSITE ORGANIZATION

Designed a composite organization based on characteristics of the interviewees' organizations.

FINANCIAL MODEL FRAMEWORK

Constructed a financial model representative of the interviews using the TEI methodology and risk-adjusted the financial model based on issues and concerns of the interviewees.

CASE STUDY

Employed four fundamental elements of TEI in modeling the investment impact: benefits, costs, flexibility, and risks. Given the increasing sophistication of ROI analyses related to IT investments, Forrester's TEI methodology provides a complete picture of the total economic impact of purchase decisions. Please see Appendix A for additional information on the TEI methodology.

DISCLOSURES

Readers should be aware of the following:

This study is commissioned by Checkmarx and delivered by Forrester Consulting. It is not meant to be used as a competitive analysis.

Forrester makes no assumptions as to the potential ROI that other organizations will receive. Forrester strongly advises that readers use their own estimates within the framework provided in the study to determine the appropriateness of an investment in Checkmarx.

Checkmarx reviewed and provided feedback to Forrester, but Forrester maintains editorial control over the study and its findings and does not accept changes to the study that contradict Forrester's findings or obscure the meaning of the study.

Checkmarx provided the customer names for the interviews but did not participate in the interviews.

The Checkmarx Customer Journey

Drivers leading to the Checkmarx investment

Interviews			
Role	**Industry**	**Region**	**Annual Revenue**
Global head of application security	Manufacturing	EMEA HQ, global operations	$180 billion+
Application security team manager	Technology	EMEA HQ, global operations	$30 billion+
Security analyst	Professional services	North America HQ, global operations	$15 billion+
Senior director of strategic technology development	Financial services	US	$8 billion
Senior director of IT operations	Healthcare	North America	$7 billion+
Application security engineer	Manufacturing	US HQ, global operations	$3 billion+
VP of engineering	Financial technology	US HQ, global operations	$70 million+
Chief enterprise architect and CISO	Technology	US	$50 million

KEY CHALLENGES

Before adopting Checkmarx, the interviewees' organizations had disjointed legacy tools and manual code review efforts across application teams. Several interviewees' organizations relied solely on manual scans conducted by SecOps or application security teams. Other organizations relied on legacy tools such as independent tools from different vendors or open-source tools that required separate installation and maintenance. One organization had multiple developer teams with siloed approaches to securing their code that leveraged either legacy solutions or manual reviews.

The interviewees noted how their organizations struggled with common challenges, including:

- **Manual scan challenges.** Before using Checkmarx, several organizations relied on manual processes to identify vulnerabilities and misconfigurations within code. These processes were prone to human error and inconsistency, reduced time to value, provided limited visibility, and ultimately made it difficult for organizations to secure their applications and IoT devices effectively. Code-scanning and validation activities were time-consuming and slow, reducing operational speed and scalability. Manual validation had limited coverage as it relied on developers and application security teams to effectively cover all possible security and misconfiguration issues without automation or visibility into vulnerability impact and root cause. This led to instances where developers did not identify or fix issues. Issues found late in the process could delay code releases.

The senior director of IT operations at a healthcare organization said: "There were a lot of pain points with code validation. Each programmer has their own way of how they script, and code validation was done by other colleagues that would sign off on any code sequences before they were reviewed by an advisory board who would review applications

before they went into production. That review took time to complete. There are a lot of manual processes tied into that to satisfy our needs."

- **High false-positive rates.** Legacy tools had high rates of false positives, requiring teams to spend time reviewing and validating flagged issues. This impacted the efficiency of security teams and slowed down development processes. The VP of engineering at a financial technology organization explained: "The biggest problem that we ran into with these kinds of tools is that they have a really high number of false positives, especially with dynamic languages. With [the prior tool], we found that there were just hundreds of false positives that we had to sift through. They were not real vulnerabilities at all, and that was really time-consuming."
- **Difficulty seeing vulnerability impact and root cause.** Interviewees noted that prior approaches lacked sufficient context for efficient remediation. Manual approaches and legacy tools provided limited visibility into root causes, vulnerability impact, and remediation paths, making it difficult for security and developer teams to quickly address issues. The VP of engineering at a financial technology organization said: "We found that [our legacy tool] was hard to operationalize. It was hard for us to see the vulnerabilities that were impacting each service on the latest version."
- **Lack of support for coding languages, frameworks, and libraries.** Legacy tools lacked coverage for specific coding languages, frameworks, and libraries that organizations used. The application security team manager at a technology organization shared that their organization had custom frameworks and libraries that, by default, were not recognized by its prior scanning tools. The VP of engineering at a financial technology organization noted that their legacy tool lacked plug-ins for some of the programming languages its developers used.
- **Disjointed approaches impeded operational speed and scalability.** With multiple legacy tools and inconsistent manual validation efforts, vulnerability management was a convoluted process for the interviewees' organizations. Prior approaches were prone to error and inconsistency across developer and application security teams, impacting the efficiency of the software development lifecycle. These approaches impacted speed, scalability, and time to value for the organizations' applications and were unsustainable in the long term.

> **"We had multiple tools running in multiple regions. We had challenges with speed and scalability. Some applications took 16 to 18 hours to scan. We needed to speed up, adopt agile, and move faster. That's why we chose Checkmarx."**
>
> *Global head of application security, manufacturing*

INVESTMENT OBJECTIVES

The interviewees' organizations searched for a solution that could:

- Harden application security risk posture and improve coding practices to reduce vulnerabilities over time.
- Consolidate application security into a central platform to drive total cost of ownership (TCO) savings and reduce complexity.

- Automate code reviews for developers and SecOps personnel.
- Provide developers with tooling integrated into continuous integration/continuous delivery (CI/CD) pipelines, integrated development environments (IDEs), and ticketing tools.
- Remediate flaws faster.

Key Assumptions

- **$10 billion in annual revenue**
- **1,000 developers**
- **1,000 applications**

COMPOSITE ORGANIZATION

Based on the interviews, Forrester constructed a TEI framework, a composite company, and an ROI analysis that illustrates the areas financially affected. The composite organization is representative of the eight interviewees, and it is used to present the aggregate financial analysis in the next section. The composite organization has the following characteristics:

Description of composite. The composite is a global organization headquartered in the US with $10 billion in annual revenue. The company has 1,000 developers who develop, update, and/or maintain 1,000 applications annually. Before Checkmarx, the composite used disjointed legacy tools and manual code checks across application development teams.

Deployment characteristics. The composite organization adopts the Checkmarx One platform to consolidate its application security solutions and retire legacy tools and manual efforts. Developers at the composite leverage Checkmarx's insights within its IDE or code repository to identify and remediate vulnerabilities and misconfigurations. Application security analysts use Checkmarx to assess code security before it is pushed to production environments.

Analysis Of Benefits

Quantified benefit data as applied to the composite

Total Benefits						
Ref.	**Benefit**	**Year 1**	**Year 2**	**Year 3**	**Total**	**Present Value**
Atr	Improved developer efficiency	$1,228,500	$1,374,750	$1,521,000	$4,124,250	$3,395,725
Btr	Improved security analyst efficiency	$393,496	$457,648	$521,800	$1,372,945	$1,127,982
Ctr	Material breach risk reduction savings	$499,698	$599,638	$699,578	$1,798,914	$1,475,443
Dtr	Technology cost consolidation	$453,600	$453,600	$453,600	$1,360,800	$1,128,036
	Total benefits (risk-adjusted)	$2,575,295	$2,885,636	$3,195,978	$8,656,908	$7,127,186

IMPROVED DEVELOPER EFFICIENCY

Evidence and data. Interviewees told Forrester that Checkmarx helped their organizations' developers improve efficiency by providing visibility into vulnerabilities and misconfigurations earlier in the development lifecycle. With Checkmarx, developers gained insight into potential security flaws during development, enabling them to fix issues before they could cause delays, disrupt development workflows, and become more time-consuming to remediate. Further, developers adopted secure coding practices by embedding security checks into development workflows, improving code security over time. With Checkmarx, developers increased their code quality and productivity through:

- **Avoided manual code reviews.** Interviewees shared that Checkmarx automatically flagged vulnerabilities during the coding process, enabling developers to avoid time-consuming and inconsistent manual review work. Additionally, developers no longer needed to wait for application security teams to conduct their code reviews and identify vulnerabilities or misconfigurations, driving further efficiency for developers.
- **Fewer false positives.** Interviewees reported that Checkmarx flagged fewer false positives than their prior tools, reducing unnecessary efforts to investigate and validate incorrect alerts. For example, the VP of engineering at a financial technology organization noted that Checkmarx reported 50% to 70% fewer false positives than its prior scanning tool, saving 10 to 20 hours per month across developer teams.
- **Reduced MTTR through richer context.** Interviewees also shared that Checkmarx provided richer context into vulnerabilities and misconfigurations, enabling developers to improve mean time to remediate (MTTR). With detailed reporting and analysis on identified vulnerabilities, including information on their location and severity, plus remediation guidance, developers could understand the root cause and impact more easily, enabling them to prioritize and quickly resolve the most critical issues. The application security engineer at a manufacturing organization reported a 50% reduction in time to discover, assess, report, and remediate risks.
- **Avoided rework efforts.** Interviewees noted that with more secure coding checks built earlier into development lifecycles, fewer issues were

pushed into production where diagnosing and fixing issues is more difficult. This reduced rework for developers. The senior director of IT operations at a healthcare organization shared that Checkmarx improved the stability and security of the organization's applications, preventing vulnerabilities from being pushed into production that would require remediation efforts.

"Checkmarx is reducing our mean time to remediate, which is a key KPI for application security. It's helping developers understand risk better and empowering them to have the knowledge to fix vulnerabilities and do it right the first time around instead of having to go back in [and fix them] after the fact."

Application security engineer, manufacturing

- **Improved coding security over time.** Interviewees noted that by integrating vulnerability identification and management into the development process, developers learned secure coding practices over time, leading to improved code quality and greater efficiency. Checkmarx provided developers with a faster feedback loop, helping them to learn during the flow of their coding work. Over time, this helped developers enhance code quality and prevent flaws that would have required remediation, driving long-term efficiency.

  The chief enterprise architect and CISO at a technology organization said: "It finds issues that developers wouldn't take the time to look for before. But now, because of this tool that does a pretty good job, they'll actually take the time to look through these things because they know they're hardening the product over time, and they're going to find less and less. It's to their advantage to start using it because over time they'll likely find fewer mistakes — and for the ones they do find, they'll have the savvy or the expertise now to fix [them]."

"It's freeing up capacity for software developers to be more productive. There are fewer issues being "pencil whipped" and pushed through to production than before because a lot of the simple security and code errors that Checkmarx identifies are no longer being found at the SecOps standpoint. They're actually being found at the DevOps standpoint. From that standpoint, there's a significant savings in that way, too."

Senior director of IT operations, healthcare

Modeling and assumptions. Forrester assumes the following for the composite organization:

- The organization employs 1,000 developers. Each spends 5% of their time on security and code reviews in the legacy environment, equating to 104 hours per year.

- Checkmarx increases developer productivity for security and code reviews by 40%, 45%, and 50% in Year 1, Year 2, and Year 3, respectively, as Checkmarx adoption widens and developers become more proficient in developing secure code.
- Developers recapture 50% of time savings into other productive tasks.
- The fully burdened hourly rate for developers is $65. This includes a burdened rate multiplier of 1.35.

Risks. Forrester recognizes that these results may not be representative of all experiences. The impact of this benefit will vary depending on:

- The number of developers and the time they previously spent on security and code reviews.
- Existing tools and code review processes.
- Hourly rates per developer role.
- Productivity recapture.

Results. To account for these risks, Forrester adjusted this benefit downward by 10%, yielding a three-year, risk-adjusted total PV (discounted at 10%) of $3.4 million.

Improved Developer Efficiency

Ref.	Metric	Source	Year 1	Year 2	Year 3
A1	Developer FTEs	Composite	1,000	1,000	1,000
A2	Annual hours spent on security and code review per developer in legacy environment	2,080*5%	104	104	104
A3	Increased developer productivity for security tasks with Checkmarx	Interviews	40%	45%	50%
A4	Annual hours saved per developer with Checkmarx	A2*A3	42	47	52
A5	Productivity recapture	TEI standard	50%	50%	50%
A6	Developer fully burdened hourly rate	TEI standard	$65	$65	$65
At	Improved developer efficiency	A1*A4*A5*A6	$1,365,000	$1,527,500	$1,690,000
	Risk adjustment	↓10%			
Atr	Improved developer efficiency (risk-adjusted)		$1,228,500	$1,374,750	$1,521,000
	Three-year total: $4,124,250			**Three-year present value: $3,395,725**	

IMPROVED SECURITY ANALYST EFFICIENCY

Evidence and data. Interviewees told Forrester that Checkmarx helped application security teams conduct code security reviews and more efficiently remediate issues. Application security teams at the interviewees' organizations leveraged Checkmarx to scan, test, and review code; detect vulnerabilities; and utilize its insights to fix security issues in collaboration with developers. With Checkmarx, application security analysts improved efficiency in the following ways:

- **Saved time on code scans and reviews.** Interviewees revealed that Checkmarx significantly expedited code scans compared to manual processes and legacy tools. The security analyst at a professional services organization highlighted that scans could be completed 50% faster with Checkmarx compared to manual scanning approaches. Similarly, the global head of application security at a manufacturing organization said moving from its legacy scanning tools to Checkmarx resulted in a time reduction of 4 hours per scan, saving thousands of hours for their security teams on an annual basis. With faster scans, customers could integrate scanning into developer workflows instead of being run asynchronously.

Improved application security engineer productivity

30% to 40%

- **Reduced time reviewing vulnerabilities and misconfigurations with developers.** In addition to flagging vulnerabilities, Checkmarx also provided security teams with contextual information and resolution advice for each vulnerability, providing application security personnel with better visibility into root cause and more easily communicate with developers for resolution. The application security engineer at a manufacturing organization explained: "Checkmarx will identify the vulnerability and pinpoint its exact location within the code base that it has been detected from. It will then show the best fix location and where to fix the root cause of the problem. In my opinion, there is not much else [Checkmarx] could do to set app security to be in a better position to remediate the vulnerability."

> **"Once it recognizes a vulnerability, it also provides visibility into the variables where the vulnerability is traveling to. There is a pictographic description that you can look directly into, whereas if you do it manually, you have to understand the whole code, subcode, and the inheritance of where the variable is located and what the value is passed on to. All of that would take a significantly longer amount of time compared to Checkmarx."**
>
> *Security analyst, professional services*

- **Reduced issues to address.** Interviewees said integrating vulnerability management into the development workflow with Checkmarx reduced the number of vulnerabilities and

misconfigurations for security teams to address. This not only drove time savings but also lessened the risk of issues being pushed to production. The senior director of IT operations at a healthcare company cited a 30% to 40% time savings for its security teams, sharing: "Developers are remediating issues instead of waiting for SecOps to conduct code reviews. They're fixing those problems on the front end before it reaches the SecOps team and before it goes into a production environment."

- **Simplified reporting for compliance and audits.** Checkmarx provided the interviewees' organizations with reporting capabilities for compliance and audits, saving time for security teams. The security analyst at a professional services organization cited that it used to take 5 to 6 hours to prepare reports with its prior manual processes, whereas with Checkmarx reporting, it was "virtually instantaneous." Similarly, the application security engineer at a manufacturing organization said that previously, it had taken days to prepare reports for leadership or auditors, but with Checkmarx, the process now only requires a few hours.

Modeling and assumptions. Forrester assumes the following for the composite organization:

- The composite has 12 application security engineers.
- Checkmarx improves application security engineer productivity by 30%, 35%, and 40%, in Year 1, Year 2, and Year 3, respectively, as Checkmarx adoption widens, and developers and security professionals become more proficient with the platform.
- The fully burdened annual salary for a security engineer is $148,500.
- Application security engineers recapture 80% of time savings to apply to other productive tasks.
- Application security engineer spend 192 hours annually creating and reviewing audit and compliance reports in the prior environment.
- Checkmarx reduces effort to create audit and compliance reports by 87%, equating to an annual savings of 167 hours.

Risks. Forrester recognizes that these results may not be representative of all experiences. The impact of this benefit will vary depending on:

- The number of application security engineers.
- Application security engineer salaries.
- Productivity recapture.
- Audit and compliance reporting requirements.

Results. To account for these risks, Forrester adjusted this benefit downward by 10%, yielding a three-year, risk-adjusted total PV of $1.1 million.

Improved Security Analyst Efficiency					
Ref.	**Metric**	**Source**	**Year 1**	**Year 2**	**Year 3**
B1	Application security engineer FTEs	Composite	12	12	12
B2	Improved application security analyst productivity with Checkmarx	Interviews	30%	35%	40%
B3	Application security engineer fully burdened annual salary	TEI standard	$148,500	$148,500	$148,500
B4	Productivity recapture	TEI standard	80%	80%	80%
B5	Subtotal: Improved application security engineer productivity	B1*B2*B3*B4	$427,680	$498,960	$570,240
B6	Hours required to create and review audit and compliance reports in legacy environment	Composite	192	192	192
B7	Reduction in hours spent creating audit and compliance reports with Checkmarx	Interviews	87%	87%	87%
B8	Avoided hours to create and review audit and compliance reports with Checkmarx	B6*B7	167	167	167
B9	Subtotal: Improved audit and compliance reporting efficiency	B8*(B3/2,080)*B4	$9,538	$9,538	$9,538
Bt	Improved security analyst efficiency	B5+B9	$437,218	$508,498	$579,778
	Risk adjustment	↓10%			
Btr	Improved security analyst efficiency (risk-adjusted)		$393,496	$457,648	$521,800
	Three-year total: $1,372,945			**Three-year present value: $1,127,982**	

MATERIAL BREACH RISK REDUCTION SAVINGS

Evidence and data. Checkmarx reduces the risk of a data breach, improving over time, by allowing developers to identify and resolve misconfigurations and vulnerabilities early in the software development lifecycle (SDLC) and helping improve secure coding practices over time. With Checkmarx, organizations reduced the risk of breach in the following ways:

- **Automated risk detection and mitigation.** By shifting security testing from a manual activity to an automated process, the interviewees' organizations improved testing coverage and identified and addressed vulnerabilities earlier in the SDLC, which enhanced the overall security of their applications. The global head of application security at a manufacturing organization said: "We moved from manual risk mitigation to being automated. We are shifting left with automation and enabling applications to be scanned by Checkmarx at a faster speed. This gives devs the ability to quickly go back and tackle issues."

Reduced likelihood of a breach with Checkmarx

35%

- **Detection of vulnerabilities.** Checkmarx helped the interviewees' organization's detect vulnerabilities and weaknesses that could be exploited by attackers if pushed to production. Additionally, Checkmarx provided risk prioritization capabilities and remediation guidance, helping developers and security teams efficiently prioritize risk mitigation efforts and address risks. The application security engineer at a manufacturing organization shared that identifying vulnerabilities was "half the battle" for application security and reported a 30% to 40% reduction in risk through the detection capabilities provided by Checkmarx.
- **Hardening applications over time and secure coding practices.** Interviewees shared that their organizations strengthened the security of applications and their resilience against potential threats or attacks over time. With the intelligence provided by Checkmarx, interviewees' organizations closed weak spots within their applications, reducing their overall attack surface. Additionally, by shifting left with Checkmarx, developers improved their knowledge of secure coding practices, reducing the risk of new vulnerabilities being introduced into code over time.

> **"It's obviously the right decision to get developers using Checkmarx and start driving change in reducing risk. We can never completely reduce the risk to zero, but we can definitely start closing some of those holes in our systems and reducing the overall threat landscape of our application portfolio, so we're not subjected to potential data breaches."**
>
> *Application security engineer, manufacturing*

Modeling and assumptions. Forrester assumes the following for the composite organization:

- The likelihood of experiencing one or more breaches that pose a material threat yearly is 89%.[3]
- The cumulative annual cost of all breaches is more than $5.7 million, which can include response and remediation costs, efforts to notify affected parties, regulatory fines, customer lawsuits, downtime, and more.[4] This figure is rightsized to the composite organization's specific characteristics such as annual revenue.
- External attacks account for 49.1% of breaches.[5]
- With Checkmarx, the reduced likelihood of a data breach is 25%, 30%, and 35% in Year 1, Year 2, and Year 3, respectively, as Checkmarx adoption increases and developers and security professionals become more proficient with Checkmarx.

Risks. Forrester recognizes that these results may not be representative of all experiences. The impact of this benefit will vary depending on:

- The maturity of an organization's application security approach in the prior environment.
- The size and scope of a security breach and the type of data compromised.
- Incident and remediation efforts.
- Business disruption and downtime caused by a breach.
- Legal and regulatory compliance costs.

Results. To account for these risks, Forrester adjusted this benefit downward by 20%, yielding a three-year, risk-adjusted total PV of $1.5 million.

> **"Checkmarx adds a layer of safety. We have found 10 to 12 legitimate vulnerabilities that we have since fixed. For example, it has found a couple of potential places vulnerable to cross-site scripting. That was a good find, and it was good to be able to fix those issues."**
>
> *VP of engineering, financial technology*

Material Breach Risk Reduction Savings

Ref.	Metric	Source	Year 1	Year 2	Year 3
C1	Likelihood of experiencing one or more breaches per year	Forrester research	89.0%	89.0%	89.0%
C2	Cumulative cost of breaches	Forrester research	$5,717,500	$5,717,500	$5,717,500
C3	Percentage of breaches originating from external attacks	Composite	49.1%	49.1%	49.1%
C4	Reduced likelihood of data breaches with Checkmarx	Composite	25%	30%	35%
Ct	Material breach risk reduction savings	C1*C2*C3*C4	$624,623	$749,547	$874,472
	Risk adjustment	↓20%			
Ctr	Material breach risk reduction savings (risk-adjusted)		$499,698	$599,638	$699,578
	Three-year total: $1,798,914		**Three-year present value: $1,475,443**		

TECHNOLOGY COST CONSOLIDATION

Evidence and data. Checkmarx enabled interviewees' organizations to reduce or eliminate spending on legacy testing solutions. The global head of application security at a manufacturing organization shared that it previously had multiple scanning tools running across multiple regions. With the adoption of Checkmarx, the organization retired several of these legacy tools, saving $400,000 to $500,000 in annual licensing, maintenance, and support costs.

Modeling and assumptions. Forrester assumes the following for the composite organization:

- The composite organization incurs $420,000 in licensing costs for legacy application security testing tools.
- The IT maintenance and support costs for the legacy tools total $84,000 annually.
- With the adoption of Checkmarx, the composite organization retires its legacy solutions, avoiding the associated annual licensing and maintenance and IT support costs.

Risks. Forrester recognizes that these results may not be representative of all experiences. The impact of this benefit will vary depending on:

- The number of legacy tools used prior to Checkmarx.
- Licensing, maintenance, and IT support per legacy solution.
- The number of legacy tools decommissioned over time.

Results. To account for these risks, Forrester adjusted this benefit downward by 10%, yielding a three-year, risk-adjusted total PV of $1.1 million.

Technology Cost Consolidation

Ref.	Metric	Source	Year 1	Year 2	Year 3
D1	Legacy licensing and technology costs in prior environment	Interviews	$420,000	$420,000	$420,000
D2	IT maintenance and support costs for legacy technology	D1*20%	$84,000	$84,000	$84,000
Dt	Technology cost consolidation	D1+D2	$504,000	$504,000	$504,000
	Risk adjustment	↓10%			
Dtr	Technology cost consolidation (risk-adjusted)		$453,600	$453,600	$453,600
	Three-year total: $1,360,800			**Three-year present value: $1,128,036**	

UNQUANTIFIED BENEFITS

- **Improved developer velocity and time to value.** Interviewees told Forrester that their organizations increased the speed at which developer teams produced new features or updates, improving overall time to value. With Checkmarx, organizations reduced manual review efforts, identified vulnerabilities and misconfigurations earlier, gained remediation guidance, and improved testing coverage to reduce the number of errors requiring remediation. These efficiencies streamlined CI/CD workflows to improve development velocity.

"A lot of the code validation and time to go into a production environment from nonproduction environment has been expedited because a lot of simplistic mistakes and vulnerabilities have been discovered within Checkmarx."

Senior director of IT operations, healthcare

- **Improved developer experience and developer training.** Interviewees shared that Checkmarx contributed to an enhanced developer experience by reducing the burdens of manual vulnerability management processes and enabling them to learn secure coding practices. Detailed insights into issues and remediation guidance saved developers time in investigating, addressing, and remediating them. Fixing issues more quickly allowed developers to spend more time working on new features and less time fixing bugs. Interviewees noted that developers favored the user interface/user experience (UI/UX) over legacy solutions, claiming it to be more modern and nonintrusive from a development workflow perspective.
- **Strong vendor support and partnership.** Interviewees mentioned that Checkmarx maintained a strong partnership and commitment to their organizations' success by providing responsive support and listening to their needs. The application security team manager at a technology company said: "We have a good relationship with Checkmarx and people in the company that listen to our issues. When we ask about something or have a request, they do everything they can do help us."

"The reality is that developers are just trying to do their job, which is complete a feature and release it to production. But security is coming in to pump the brakes and make sure the code is secure. Checkmarx is empowering the developers to have the best developer experience, giving them training, shifting as left as possible, and empowering them to have the right knowledge to fix issues and do it right the first time around, so they don't have to go back in after the fact."

Application security engineer, manufacturing

- **Improves ability to meet compliance requirements.** Interviewees highlighted that Checkmarx improved testing coverage across security standards and regulations, helping their organizations to effectively uphold compliance requirements. The senior director of IT operations at a healthcare organization shared, “It’s allowed developers to be much more proactive in meeting a lot of the criterion standards, especially in healthcare, because of the challenges you face because of HIPAA, patient health, GDPR, NIST — all that good stuff.”

FLEXIBILITY

The value of flexibility is unique to each customer. There are multiple scenarios in which a customer might implement Checkmarx and later realize additional uses and business opportunities, including:

- **Scalability.** Checkmarx positioned interviewees’ organizations to more easily accommodate future growth and software development needs. With the efficiencies enabled by Checkmarx, organizations can handle increased workloads, expand their application portfolios, and adapt to evolving security requirements with less effort and resources compared to legacy approaches. Furthermore, rolling out Checkmarx to other developer teams becomes easier.
- **Future-proofing.** Checkmarx helps the interviewees’ organizations continuously evolve and adapt to emerging security threats and trends. With regular updates and enhancements, Checkmarx ensures that their organizations can stay ahead of potential vulnerabilities and compliance requirements. This flexibility enables them to maintain a proactive security posture and minimize the risk of costly security breaches or software vulnerabilities.

Flexibility would also be quantified when evaluated as part of a specific project (described in more detail in Appendix A).

Analysis Of Costs

Quantified cost data as applied to the composite

Total Costs							
Ref.	**Cost**	**Initial**	**Year 1**	**Year 2**	**Year 3**	**Total**	**Present Value**
Etr	Checkmarx licensing costs	$0	$554,400	$554,400	$554,400	$1,663,200	$1,378,711
Ftr	Implementation, training, and ongoing costs	$489,216	$283,500	$283,500	$283,500	$1,339,716	$1,194,239
	Total costs (risk-adjusted)	$489,216	$837,900	$837,900	$837,900	$3,002,916	$2,572,950

CHECKMARX LICENSING COSTS

Evidence and data. Licensing costs for Checkmarx were based on price per user. For the interviewees' organizations, licensing costs varied depending on the number of users, type of license, and other contract terms/agreements.

Modeling and assumptions. Forrester assumes the following for the composite organization:

- The composite organization pays $528,000 in licensing costs for Checkmarx annually.
- Pricing may vary. Contact Checkmarx for additional details.

Risks. Forrester recognizes that these results may not be representative of all experiences. The impact of this benefit will vary depending on:

- The number of licensed users.
- License types.

Results. To account for these risks, Forrester adjusted this cost upward by 5%, yielding a three-year, risk-adjusted total PV (discounted at 10%) of $1.4 million.

Checkmarx Licensing Costs						
Ref.	**Metric**	**Source**	**Initial**	**Year 1**	**Year 2**	**Year 3**
E1	Checkmarx annual licensing costs	Composite		$528,000	$528,000	$528,000
Et	Checkmarx licensing costs	E1	$0	$528,000	$528,000	$528,000
	Risk adjustment	↑5%				
Etr	Checkmarx licensing costs (risk-adjusted)		$0	$554,400	$554,400	$554,400
	Three-year total: $1,663,200			**Three-year present value: $1,378,711**		

IMPLEMENTATION, TRAINING, AND ONGOING COSTS

Evidence and data. The interviewees' organizations incurred professional services and internal labor costs associated with implementing Checkmarx. Interviewees reported that developers received 2 to 10 hours of training on Checkmarx. Interviewees also shared that DevOps resources were involved in ongoing management, with the application security team manager at a technology firm reporting that their organization had two DevOps employees dedicated to ongoing management.

Modeling and assumptions. Forrester assumes the following for the composite organization:

- The composite organization pays a one-time professional services fee of $200,000 to support implementation and fine-tuning efforts.
- Internal labor dedicated to the implementation has a value of $62,400.
- Each developer participates in 3 hours of training at a fully burdened cost of $65 per hour, equating to $195,000 in total developer training costs.
- Each application security resource participates in 10 hours of training at a fully burdened cost of $71 per hour, equating to $8,520.
- Two DevOps engineer FTEs are dedicated to ongoing management of Checkmarx.
- The fully burdened annual salary of a DevOps engineer is $135,000.

Risks. Forrester recognizes that these results may not be representative of all experiences. The impact of this benefit will vary depending on:

- Professional services fees.
- The number of internal resources dedicated to implementation and their burdened costs.
- Training requirements for developers, the number of developers receiving training, and their fully burdened cost.
- Training requirements for application security engineers, the number of application security engineers receiving training, and their fully burdened cost.
- Ongoing management requirements, the number of resources dedicated to ongoing management, and their fully burdened cost.

Results. To account for these risks, Forrester adjusted this cost upward by 5%, yielding a three-year, risk-adjusted total PV of $1.2 million

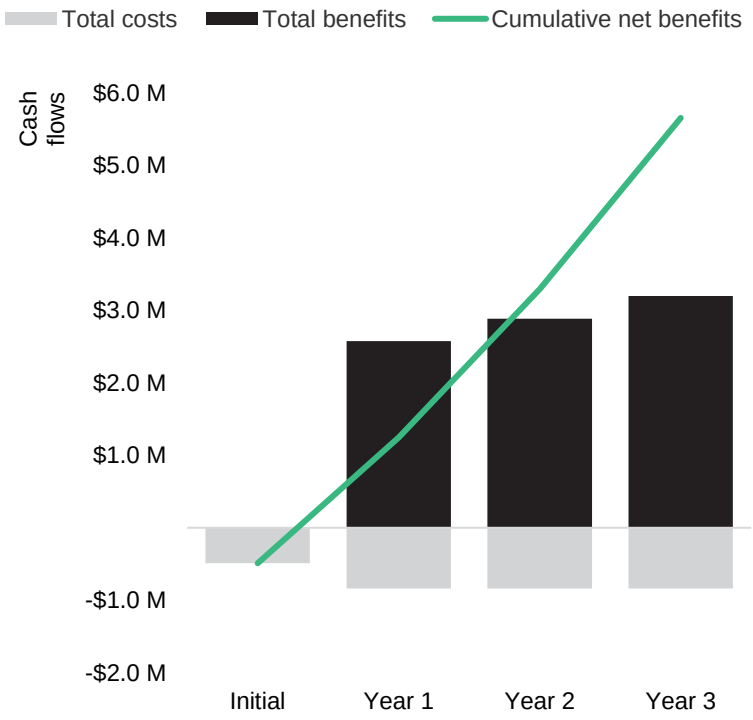
Implementation, Training, And Ongoing Costs

Ref.	Metric	Source	Initial	Year 1	Year 2	Year 3
F1	Implementation fees	Interviews	$200,000			
F2	Total developer hours for implementation	Composite	960			
F3	Developer fully burdened hourly rate	A6	$65			
F4	Internal developer labor dedicated to implementation	F2*F3	$62,400			
F5	Subtotal: Total implementation costs	F1+F4	$262,400			
F6	Developer FTEs	A1	1,000			
F7	Training time per developer (hours)	Composite	3			
F8	Training costs for developers	F3*F6*F7	$195,000			
F9	Application security engineer FTEs	B1	12			
F10	Training time per application security engineer (hours)	Composite	10			
F11	Application security engineer fully burdened hourly rate	Composite	$71			
F12	Training costs for application security engineers	F9*F10*F11	$8,520			
F13	Subtotal: Total training costs	F8+F12	$203,520			
F14	Ongoing DevOps engineer FTEs	Interviews		2	2	2
F15	Developer fully burdened rate	Composite		$135,000	$135,000	$135,000
F16	Subtotal: Total ongoing costs	F14*F15		$270,000	$270,000	$270,000
Ft	Implementation, training, and ongoing costs	F5+F13+F16	$465,920	$270,000	$270,000	$270,000
	Risk adjustment	↑5%				
Ftr	Implementation, training, and ongoing costs (risk-adjusted)		$489,216	$283,500	$283,500	$283,500
	Three-year total: $1,339,716			**Three-year present value: $1,194,239**		

Financial Summary

CONSOLIDATED THREE-YEAR RISK-ADJUSTED METRICS

Cash Flow Chart (Risk-Adjusted)



The financial results calculated in the Benefits and Costs sections can be used to determine the ROI, NPV, and payback period for the composite organization's investment. Forrester assumes a yearly discount rate of 10% for this analysis.

These risk-adjusted ROI, NPV, and payback period values are determined by applying risk-adjustment factors to the unadjusted results in each Benefit and Cost section.

Cash Flow Analysis (Risk-Adjusted Estimates)

	Initial	Year 1	Year 2	Year 3	Total	Present Value
Total costs	($489,216)	($837,900)	($837,900)	($837,900)	($3,002,916)	($2,572,950)
Total benefits	$0	$2,575,295	$2,885,636	$3,195,978	$8,656,908	$7,127,186
Net benefits	($489,216)	$1,737,395	$2,047,736	$2,358,078	$5,653,992	$4,554,236
ROI						177%
Payback						<6 months

Appendix A: Total Economic Impact

Total Economic Impact is a methodology developed by Forrester Research that enhances a company's technology decision-making processes and assists vendors in communicating the value proposition of their products and services to clients. The TEI methodology helps companies demonstrate, justify, and realize the tangible value of IT initiatives to both senior management and other key business stakeholders.

TOTAL ECONOMIC IMPACT APPROACH

Benefits represent the value delivered to the business by the product. The TEI methodology places equal weight on the measure of benefits and the measure of costs, allowing for a full examination of the effect of the technology on the entire organization.

Costs consider all expenses necessary to deliver the proposed value, or benefits, of the product. The cost category within TEI captures incremental costs over the existing environment for ongoing costs associated with the solution.

Flexibility represents the strategic value that can be obtained for some future additional investment building on top of the initial investment already made. Having the ability to capture that benefit has a PV that can be estimated.

Risks measure the uncertainty of benefit and cost estimates given: 1) the likelihood that estimates will meet original projections and 2) the likelihood that estimates will be tracked over time. TEI risk factors are based on "triangular distribution."

The initial investment column contains costs incurred at "time 0" or at the beginning of Year 1 that are not discounted. All other cash flows are discounted using the discount rate at the end of the year. PV calculations are calculated for each total cost and benefit estimate. NPV calculations in the summary tables are the sum of the initial investment and the discounted cash flows in each year. Sums and present value calculations of the Total Benefits, Total Costs, and Cash Flow tables may not exactly add up, as some rounding may occur.

PRESENT VALUE (PV)

The present or current value of (discounted) cost and benefit estimates given at an interest rate (the discount rate). The PV of costs and benefits feed into the total NPV of cash flows.

NET PRESENT VALUE (NPV)

The present or current value of (discounted) future net cash flows given an interest rate (the discount rate). A positive project NPV normally indicates that the investment should be made unless other projects have higher NPVs.

RETURN ON INVESTMENT (ROI)

A project's expected return in percentage terms. ROI is calculated by dividing net benefits (benefits less costs) by costs.

DISCOUNT RATE

The interest rate used in cash flow analysis to take into account the time value of money. Organizations typically use discount rates between 8% and 16%.

PAYBACK PERIOD

The breakeven point for an investment. This is the point in time at which net benefits (benefits minus costs) equal initial investment or cost.

Appendix B: Endnotes

[1] Source: The Forrester Wave™: Software Composition Analysis, Q2 2023, Forrester Research, Inc., June 2023

[2] Total Economic Impact is a methodology developed by Forrester Research that enhances a company's technology decision-making processes and assists vendors in communicating the value proposition of their products and services to clients. The TEI methodology helps companies demonstrate, justify, and realize the tangible value of IT initiatives to both senior management and other key business stakeholders.

[3] Source: Security Survey, 2023, Forrester Research, Inc., October 2023

[4] Ibid.

[5] Ibid.


FORRESTER®