Sponsored by

The State of the State of Application Exploits in Security Incidents

A "SO-SO" MULTI-SOURCE ANALYSIS OF PUBLISHED INDUSTRY RESEARCH

ON THE DOCKET

Analysis for this report was provided by the Cyentia Institute. Cyentia is a research and data science firm working to advance cybersecurity knowledge and practice. We do this by partnering with security vendors and other organizations to publish a range of high-quality, data-driven content like this study.

Find out more at www.cyentia.com.

F5 Labs provides credible, reliable, unbiased strategic threat intelligence to the security community using data from internal and industry sources. All research is data-driven, analyzed by industry veterans, and never promotes F5 products. Our goal is to help security professionals make risk-based decisions and support security program changes within their organizations.

Find out more at www.f5labs.com.

Introduction

FOR, QUARRELING, EACH TO THEIR VIEW THEY CLING.

SUCH FOLK SEE ONLY ONE SIDE OF A THING.

— BUDDHA

Cybersecurity is always about perspective, and that's doubly true when talking about application security. Applications constantly change and so too do the philosophies and practices used to develop and protect them. Trying to get a current reading on the state of application security is like the parable of the blind men and an elephant. Each person feels a different part of the elephant's body—the ear, the tail, the tusk—and each comes away with a completely different description of the animal. Each of those perspectives is true in their own context, but we have to acknowledge that a proper understanding of the whole requires more than studying the individual parts (whether we're talking elephants or applications).

This study is an attempt to stitch together a more complete view of the application security elephant. More specifically, we examine published industry reports from multiple sources to develop a better understanding of the frequency and role of application exploits in security incidents. Along the way, we'll demonstrate the challenges of multi-source analysis and offer recommendations on how research producers can make it easier for those who want to piece together the bigger picture.

“

It was six men of Indostan
To learning much inclined,
Who went to see the Elephant
(Though all of them were blind),
That each by observation
Might satisfy his mind.
…

And so these men of Indostan
Disputed loud and long,
Each in his own opinion
Exceeding stiff and strong,
Though each was partly in the right,
And all were in the wrong!

— John Godfrey Saxe

KEY FINDINGS

56% of the largest incidents of the last 5 years tie back to some form of web application security issue, constituting 42% of all financial losses recorded for these extreme events.

254 The average time-to-discovery for incidents involving web application exploits (254 days) is significantly higher than the 71-day average among other extreme loss events we studied.

57% of all reported financial losses for the largest web application incidents over the last five years were attributed to state-affiliated threat actors. Update threat models accordingly.

#1 Web application attacks were the leading incident pattern among data breaches for 6 of last 8 years .

Top 2 Exploit Public-Facing Application is the #1 or #2 technique for all sources that report Initial Access tactics defined in the ATT&CK framework.

12% of threat groups are known to use the ATT&CK tactic exploit public facing applications and 42% leverage valid user accounts (often via web apps) to gain initial access to target organizations.

So-So There's not much consensus among sources regarding the most common types of web application vulnerabilities and attacks. But we saw at least "so-so" agreement among them that injection attacks and cross-site scripting rank highest.

What's Multi-Source Analysis?

We started this project with the intent of conducting a meta-analysis of published research on the role of web application attacks in cybersecurity incidents. After spending far more time and effort to identify sources, extract relevant information, and analyze and compare findings, we came to the disappointing conclusion that the source material simply wasn't sufficient to support proper meta-analysis. We reoriented our approach around the concept of multi-source analysis.

True meta-analysis assumes source studies that share comparable research questions, methods, quality, and have a measurable degree of error. Multi-source analysis, on the other hand, anticipates the challenges inherent to integrating, normalizing, and analyzing data sets that differ in numerous ways.[1] Meta-analysis is difficult to impossible given the current state of security industry research. But multi-source analysis is achievable and part of our goal in this study is to demonstrate that approach.

> “ **MULTI-SOURCE ANALYSIS** anticipates the challenges inherent to integrating, normalizing, and analyzing data sets that differ in numerous ways.

What's the Cyentia Research Library?

We heavily relied on the Cyentia Research Library to find research reports to include in this multi-source analysis. We started the Library several years ago with the goal of curating published cybersecurity industry research to make it more accessible and useful for the community. It now contains over 2,500 reports on a variety of topics, including web application attacks. The Library is free and open to the community.

Both meta-analysis and multi-source analysis start with the ability to identify relevant source material. We believe these methods are essential to mature the state of research and practice in our industry, and it's our hope that the Cyentia Library will foster that goal. We thank F5 Labs for helping to make that goal a reality by sponsoring this research. If your organization would be interested in supporting the curation and expansion of the Cyentia Library, please reach out to info@cyentia.com.

[1] https://datasciencecmu.wordpress.com/2014/03/28/introduction-to-multi-source-data-analysis/

Web Application Exploits in Security Incidents

The most obvious way to gauge the state of web application security is to look at successful attacks—incidents that resulted in some kind of a loss to the company. That could mean damage to the system, an outage that disrupted company operations, or the theft of money (or cryptocurrency) or sensitive information. In order to understand such incidents, we turn to two primary sources: our own Information Risk Insights Study 20/20 "Extreme Edition" (IRIS Xtreme) and the granddaddy of them all, Verizon's Data Breach Investigations Report (DBIR). We'll also ascertain what can be seen through MITRE's popular ATT&CK project.

Extreme Cyber Loss Events

Cyentia's IRIS 20/20 Xtreme analyzes the 100 largest cyber loss events of the last 5 years, totaling $18 billion in reported losses and 10 billion compromised records. The study draws from Advisen's Cyber Loss Data, which contains nearly 100,000 cyber events collected from public sources. We supplemented that dataset by collecting hundreds of additional data points on each of these extreme cyber loss events. If you're looking for stats on the frequency of major security incidents, how much they cost, what makes them worse, who's behind them, and how they go down, then you'll definitely want to put this one on your reading list.

Based on details we were able to glean from public records[2], 30 of these mega loss events exploited published vulnerabilities (CVEs) affecting web applications and/or incorporated application-layer attacks like SQL injection or cross-site scripting. What's more, these 30 incidents combined to rack up $5.7 billion in verifiable losses. That ranks web application exploits among the most common and damaging contributors to extreme cyber loss events.

If we extend our definition of web application security issues beyond just exploitation of code-level vulnerabilities to include attacks targeting authentication credentials (most do), the numbers get even bigger. Overall, this was the most common (46 incidents) and costly ($10 billion) tactic observed across the extreme loss events we studied. But if we just count those involving web application credentials, we're left with 32 incidents (still the most common) at a tad more modest cost of $6.5 billion. Most of these attacks took advantage of weak or stolen passwords but some leveraged fancier techniques such as credential stuffing.

Interesting Fact: The average time-to-discovery for web application exploits was 254 days, which is significantly higher than the 71 days that was the average of all 103 extreme events in the IRIS-X.

[2] We view this as a conservative amount because it's nigh certain that some additional number of the extreme events we studied involved application exploits that were not known or publicized in our source material.

But wait—there's more! Many web application security frameworks include configuration errors that expose sensitive data (e.g., OWASP A3:2017). We found 13 of those among our list of extreme events but, curiously, they had no public record of financial losses. That's probably due to the nature of such incidents; huge amounts of data are exposed but not provably compromised by malicious parties. Thus, they tend to fall through the cracks of loss-amplifying consequences like regulatory penalties and class-action lawsuits.

Keep in mind that the categories we just went through aren't mutually exclusive. In other words, the same event might utilize application exploits and credential attacks. So, you can't simply sum all the numbers we shared above. But don't fret; we're as interested in the grand total as you are. All told, 58 of the 103 biggest incidents of the last 5 years (56%) tie back to some form of web application security issue. With an all-inclusive price tag of $7.6 billion and some change, this constitutes 42% of all financial losses recorded for these extreme events.

> “ All told, 58 of the 103 biggest incidents of the last 5 years (and 42% of all associated financial losses) tie back to some form of web application security issue.

The stats shared above are sufficient to establish that web application security plays a major role in major cyber loss events. But is that all we can glean from a study analyzing the worst of the worst incidents in recent history? We don't think so; let's see what else we can discern by looking through the IRIS-X.

Many corporate risk registers we've seen over the years report on cyber risk by grouping common actors, actions, technical impacts, etc., into broad incident patterns or scenarios. We adopted a similar approach for the IRIS-X and came up with 8 major incident patterns. Only 4 of them show up among the 58 web application security incidents, however. Those are listed along with their associated frequency and combined losses in Figure 1.

We already mentioned web application hacks and exposed data stores, so this categorization doesn't add much. The inclusion of cryptocurrency theft may come as a surprise, though, because it's a niche risk that falls outside the threat model of many organizations.

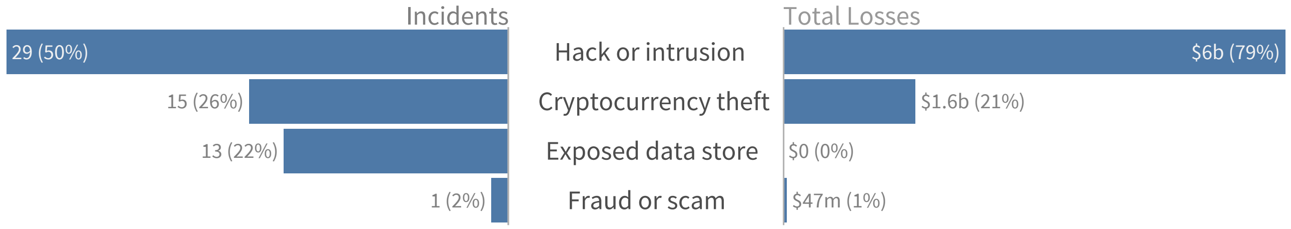


FIGURE 1: FREQUENCY AND LOSSES FOR BROAD INCIDENT PATTERNS ASSOCIATED WITH EXTREME WEB APPLICATION INCIDENTS

The cryptocurrency theft pattern includes all attacks on exchanges or the underlying cryptocurrency systems themselves. We found way more of them than expected, both for the IRIS-X and for this breakout of web application security incidents. But when you think about how these exchanges work—tons of money (or something people value as money) accessible via (sometimes surprisingly sketchy) web applications—it starts to make sense. After studying these incidents, the best advice we can give is a) choose a reputable exchange, b) enable multi-factor authentication, and c) store your cryptocurrency in cold or hardware wallets.

Let's switch over to the actors behind these events in Figure 2. Based on the incident patterns in Figure 1, you can probably guess that the majority originate from threat actors outside the victim organization. In fact, the accidental data store exposures are the only insider incidents of the bunch. That's a good reminder that insider risk encompasses much more than employees acting maliciously.

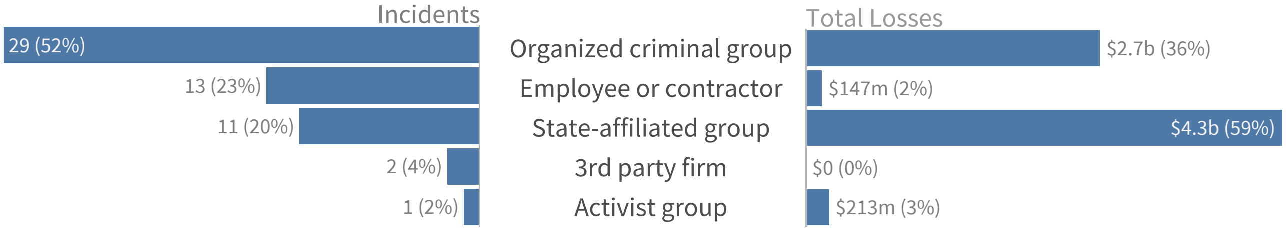


FIGURE 2: FREQUENCY AND LOSSES FOR ACTORS TIED TO EXTREME WEB APPLICATION INCIDENTS

We've come to expect cybercriminal gangs to drive a large proportion of incidents, so it's not surprising to see them behind web application attacks as well. To our mind, the real story here is that almost one in five events is attributed to state-affiliated actors. Even more surprising—these state actors caused $4.3B in damages. That constitutes 57% of all known losses for the largest web application incidents over the last five years! Nation-state attacks might not be common in the grand scheme of cybersecurity incidents, but they account for a large portion of tail risk.

> “
> **THE REAL STORY HERE** is that almost one in five events is attributed to state-affiliated actors. Even more surprising—these state actors caused $4.3B in damages.

Having lined up the suspects indicted for extreme loss events related to web applications, we'll now investigate the tactics employed to pull off these crimes. "Investigate" might be a little misleading here, because we don't have any forensic evidence to sift through for clues. But we did scour public sources to piece together as much as possible and were generally able to confirm some specific VERIS-based threat actions for each incident. Those are recorded in Figure 3.

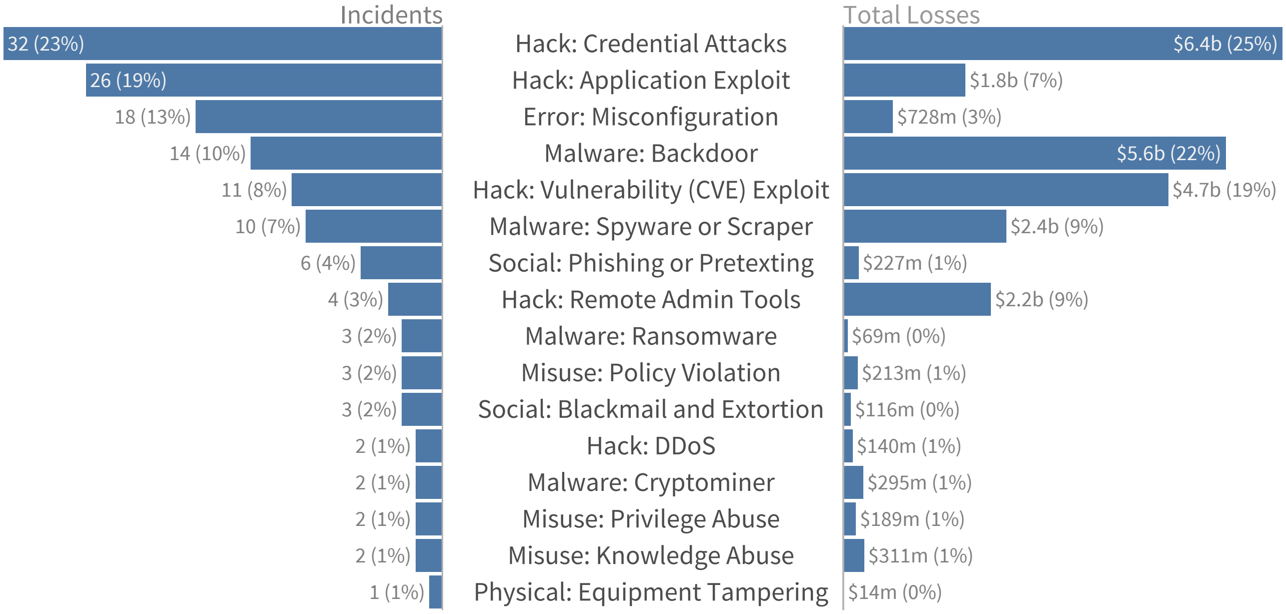


FIGURE 3: FREQUENCY AND LOSSES FOR THREAT ACTIONS IDENTIFIED IN EXTREME WEB APPLICATION EVENTS

There's a lot we could say about Figure 3, but this section is already growing long. Plus, the data speaks pretty well for itself. Allow us to interpret the gist of it: Fix your code; Patch your systems; Double-up your creds; Watch your back(door). Oh yeah—and add some policies and procedures to lock down those web-facing data stores.

That's all common knowledge, of course. But apparently not quite so common in practice. Use these results as motivation to that end.

TLDR: Web application security is a really big deal among really big incidents. You probably already knew that, but now you have some scary-yet-legit numbers for slideware, business cases, and lively discussion around the family dinner table.

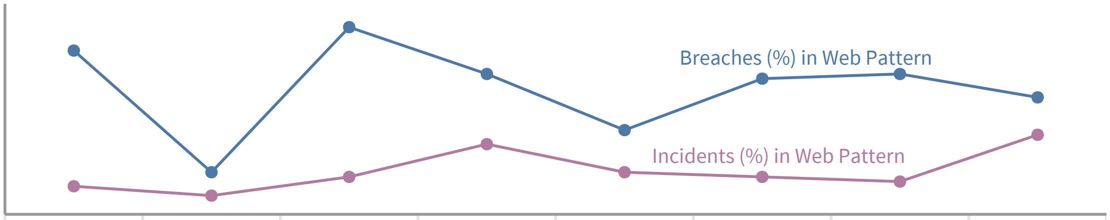
General Incidents & Breaches

We learned from the IRIS-X that web applications play a big role in the worst-of-the-worst security incidents, but what about more "regular" events? For gaining that perspective, there's no better source than Verizon's Data Breach Investigations Report. The DBIR analyzes tens of thousands of security incidents from scores of diverse sources each year. Undoubtedly, some of those incidents are the same extreme events featured in IRIS-X, but the majority of them are the kinds of things organizations experience on a regular basis. Afterall, the 2021 DBIR started with 79,635 incidents reported in 2020; that's 218 for each day of the year!

While the IRIS-X relies on public sources for details on incidents, the DBIR gets information straight from leading incident response (IR) firms, government agencies, law enforcement agencies, intel sharing organizations (ISACs/ISAOs), cybersecurity vendors, and other first-hand sources. No other report even comes close to amassing information at this scale and depth in one place. Thus, the DBIR is a meta-analysis in itself. For our own meta-analysis, we'll focus on the meta-analysis from the latest (2021) DBIR but also retrieve stats going back to 2014.[3]

For several years now, the DBIR has categorized incidents into incident classification patterns. Think of these patterns as incidents that share common types of threat actors, actions or methods, assets, and technical impacts. The DBIR uses these patterns as a way to distill the complexity of categorizing thousands of incidents into something we can all get our heads around. And it does that effectively; the 8 patterns presented in the 2021 report cover 99.6% of all incidents reported for the year.

Conveniently, the DBIR has a dedicated incident pattern for web application attacks. That alone signals that such events are important/common enough to make the Elite Eight. The DBIR defines them as "simple web application attacks with a small number of steps/additional actions after the initial web application compromise." There were 4,862 incidents (17% of all incidents) meeting this definition in the 2021 report and 1,384 were confirmed data breaches (26% of all breaches).[4] That ranks web application attacks #2 for both incidents and breaches (see rightmost column of Figure 4).[5]



	2014	2015	2016	2017	2018	2019	2020	2021
Breaches (pct)	35%	9%	40%	30%	18%	29%	30%	25%
Breaches (rank)	1	5	1	1	1	1	1	2
Incidents (pct)	6%	4%	8%	15%	9%	8%	7%	17%
Incidents (rank)	6	5	7	4	4	5	4	2

FIGURE 4: HISTORICAL PROPORTION OF INCIDENTS AND DATA BREACHES CLASSIFIED AS WEB APPLICATION ATTACKS IN THE VERIZON DBIR

[3] This is all getting so meta...

[4] The DBIR distinguishes incidents (any security-compromising event) from breaches (incidents involving confirmed data disclosure).

[5] It should be noted that some of the other categories (e.g., DDoS) contain application-layer attacks. But they're not simple to extract and so we'll keep our focus on the named Basic Web Application Attacks pattern. Just keep in mind that the total frequency of all application attacks is even higher.

The patterns were reconstituted for the 2021 DBIR, so it's not quite apples-to-apples to compare across prior reports. But it's not unreasonable to do so for web application attacks, since the vast majority of incidents and breaches remain consistent for that category.[6] Figure 4 makes this comparison, going all the way back to 2014 when the incident classification patterns first emerged. From this, we see that web application attacks are generally the most common pattern among data breaches and fall middle of the pack among all reported security incidents.

Another nifty thing the DBIR does each year is break these incident patterns out by sector (and sometimes by organization size and region). We're not going to repeat all that here, but we do think it fits with the task at hand to observe that web application attacks rank in the top 3 patterns for 14 of 16 sectors. It's the most prevalent incident pattern for 5 sectors.

Diving deeper into the actors and actions behind web application attacks in the DBIR reveals findings that mirror what we saw in the IRIS-X. Well, at least they do if we remove the exposed data store events, which fall under the DBIR's Miscellaneous Errors pattern. Web application attacks originate almost entirely from external actors and weak or stolen passwords reign as leading contributors, followed by vulnerability exploits and then backdoors. So, once again: Fix your code; Patch your systems; Double-up your creds; Watch your back(door).

Another Source of Corroborating Evidence

The DBIR might be the biggest kid on the block when it comes to breach reports, but it's not the only one. The Trustwave Global Security Report is another long-running analysis of forensic investigations. The 2020 report shows that 11% of incidents investigated by Trustwave involved application exploits, ranking as the second most common factor contributing to compromise (see page 19).

Based on our interpretation from the Trustwave report, it appears that their category of application exploits doesn't exactly align with the patterns as defined by the DBIR. The DBIR classifies code injection, weak passwords, and credential stuffing under the basic web application attacks pattern, but those are listed separately by Trustwave. The point isn't who's right or wrong here. We bring this up because 1) it narrows the discrepancy between the two reports with respect to the percent of incidents tied to web application attacks and 2) it demonstrates the "loose translation" that is necessary to meta-analysis when there's no common language. Both reports agree that web applications are a major target in real-world security incidents.

TLDR: In addition to being a big deal for the biggest loss events, web app attacks are among the most common patterns among all manner of security incidents and breaches. Cracked, stolen, and reused passwords along with vulnerability exploits are popular tactics. If they're not high on your security radar, you might wanna have that thing checked.

[6] See Figures 47 and 48 in the 2021 DBIR for a numerical comparison of old-to-new patterns.

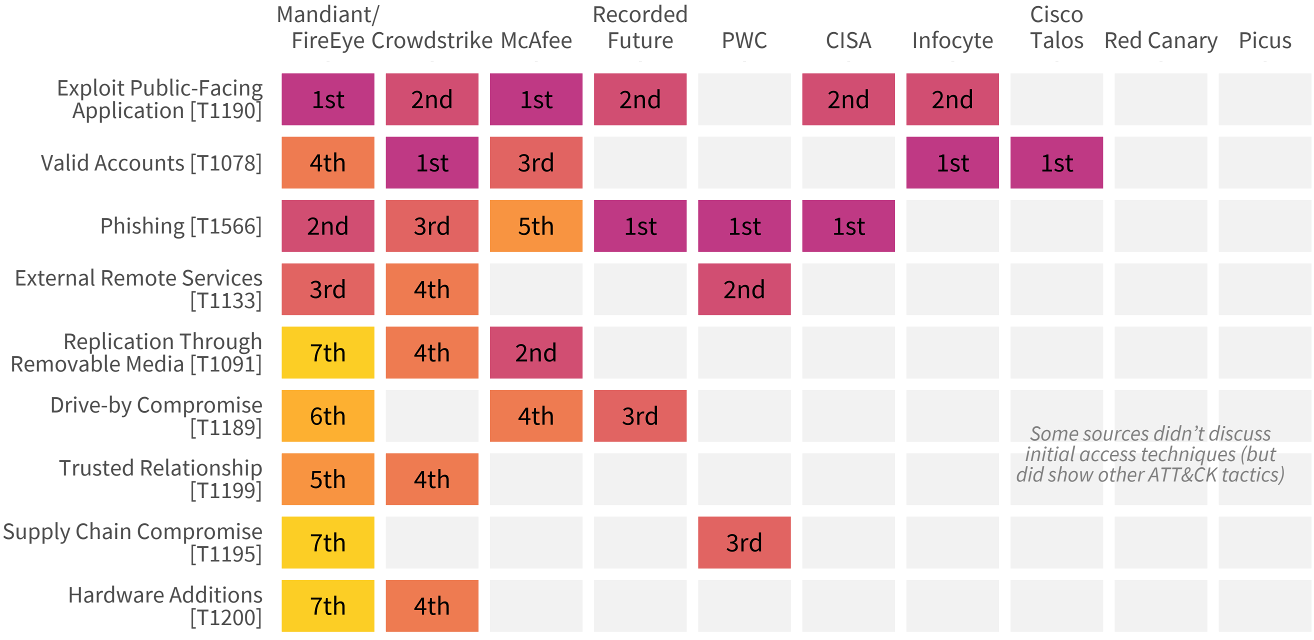
Web Application ATT&CKs

We were able to make meaningful comparisons between the IRIS-X and DBIR because they share a common vocabulary in VERIS. But that's not the case for most cybersecurity research reports, which tend to all speak their own language. MITRE ATT&CK seems to be changing that state of affairs, however.

ATT&CK is a knowledge base of adversary tactics and techniques based on real-world observations. It's free for use by any organization and has gained a lot of traction over the last few years. Due to this popularity, a growing number of industry research reports present findings based on ATT&CK. Since web application exploits are a recognized ATT&CK technique (T1190), we wanted to investigate the industry consensus (or lack thereof) around the frequency of this attack vector relative to others used to gain Initial Access into target environments.

Step one, of course, was to find sources that use ATT&CK as the basis for reporting incident details. The Cyentia Research Library got us started down that road and we Googled the rest of the way until we kept arriving at the same sources. Undoubtedly, we missed some; apologies. Sources we found are shown in Figure 5 and referenced more fully in the Appendix.

With reports rounded up, the next step was extracting relevant data points. We realized very quickly that the sources varied substantially in terms of the ATT&CK techniques they observed and reported. Much of that variation ties back to how each source collects data. Full-service IR firms, for example, tend to gain broader visibility across the spectrum of ATT&CK techniques than a point solution product vendor.



	Mandiant/ FireEye	Crowdstrike	McAfee	Recorded Future	PWC	CISA	Infocyte	Cisco Talos	Red Canary	Picus
Exploit Public-Facing Application [T1190]	1st	2nd	1st	2nd		2nd	2nd			
Valid Accounts [T1078]	4th	1st	3rd				1st	1st		
Phishing [T1566]	2nd	3rd	5th	1st	1st	1st				
External Remote Services [T1133]	3rd	4th			2nd					
Replication Through Removable Media [T1091]	7th	4th	2nd							
Drive-by Compromise [T1189]	6th		4th	3rd						
Trusted Relationship [T1199]	5th	4th								
Supply Chain Compromise [T1195]	7th				3rd					
Hardware Additions [T1200]	7th	4th								

Some sources didn't discuss initial access techniques (but did show other ATT&CK tactics)

FIGURE 5: TOP INITIAL ACCESS TECHNIQUES ACCORDING TO MULTIPLE SOURCES

The result of this variation in visibility is what initially seems to be an utter lack of consensus among sources as to which ATT&CK techniques are most common. But what we're really seeing is disparate reporting from various sources because they do different things at different points in the progression of tactics. It has nothing to do with the frequency of web application attacks. Because of that, we decided to focus strictly on the Initial Access tactic under which the technique of Exploit Public-Facing Application falls. This more appropriately calibrates the scale for what we really want to measure.

The results in Figure 6 show surprising strong consensus, given where we started. Exploit Public-Facing Application is the #1 or #2 technique for all sources that report it. Yes, there are a lot of missing values and it's true we don't know whether that's because related techniques couldn't be observed or simply weren't observed. But multi-source analysis is messy. Especially when we're not dealing with controlled experiments and clearly defined methodologies. We were able to draw useful insight that corroborates what we saw in IRIS-X and the DBIR, and that's a win in our book.

Threat Actors Known to Exploit Web Applications

Another nifty thing MITRE's doing with ATT&CK is consolidating intelligence on tactics and techniques used by various adversaries. Rather than figuring out each vendor's nomenclature for an adversary of interest and then gathering a ton of disparate information released across a plethora of advisories, reports, blog posts, and tweets, you can just swing by the Groups section of ATT&CK where a lot of that work's been done for you.

We haven't attempted any assessment as to the completeness of this information collated by MITRE[7], but let's just take it at face value for the moment. We're curious to know what proportion of threat groups exploit web applications and how that compares with other Initial Access techniques (including sub-techniques). We scooped up the data from the ATT&CK website and used it to create Figure 6.

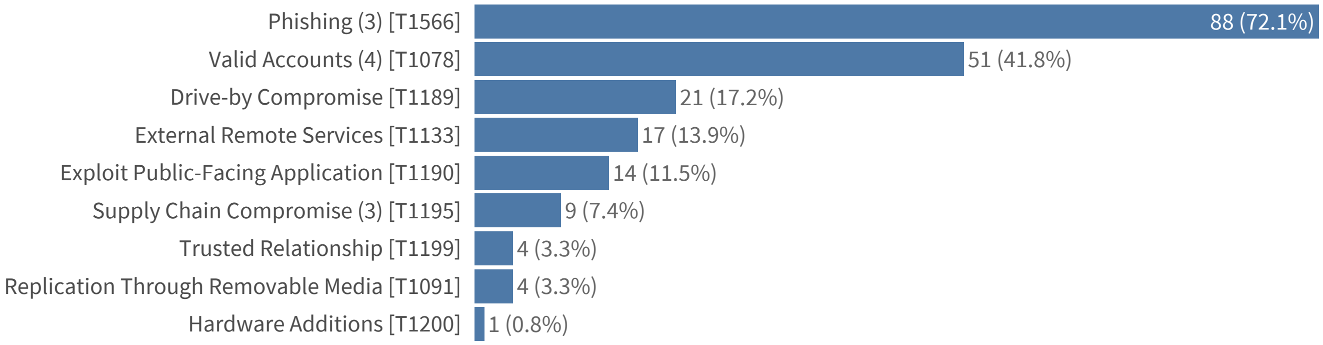


FIGURE 6: INITIAL ACCESS TECHNIQUES BASED ON KNOWN USAGE AMONG THREAT GROUPS

According to this way of parsing things, web application attacks fall right in the middle of other initial access techniques. Just 14 of the 122 (11.5%) threat groups tracked by MITRE are known to exploit web applications as a vector into target organizations. That seems low to us, but we'll continue with a willing suspension of disbelief here. After all, there's no reason to think it's any more or less complete for this technique than others in Figure 6.

Figure 6's ranking isn't quite the top billing received in previous sections, but enough to conclude that popping web apps isn't seen as unfashionable amongst the baddies. To that point, we can't help but wonder at the contrast between Figure 5 and Figure 6. Namely, why isn't phishing more commonly observed and reported? But that's not the focus of our current investigation, so we won't dwell on it for now. Ain't it fun how the results of one bit of analysis open up new things worth looking into?

TLDR: Analysis of multiple sources that report stats on ATT&CK tactics corroborates what we learned from the IRIS-X and DBIR: web application exploits are among the most common techniques observed in security incidents. Threat actors like them too.

[7] But Cyentia operators are standing by for any callers wishing to sponsor such an investigation.

Web Application Attacks & Vulnerabilities

In taking on this project, we were mostly interested in studying how web applications exploits factored into confirmed security incidents. But it became apparent in scouring source material that reports more commonly present statistics on attacks (that aren't necessarily successful or result in incidents) and vulnerabilities (that may never be attacked or exploited) than incidents. We also learned that these sources and statistics range widely in terms of scope, methods, quality, etc., making it a real challenge to synthesize findings across them. So, we've decided to add this short section to briefly summarize the trials, tribulations—and to a lesser extent—takeaways about web application attacks and vulnerabilities.

Common Web Application Attacks

What types of web application attacks are most common? It's a reasonable question, but not so easy to answer. Setting aside the fundamental question about what constitutes an *attack*, no single source sees everything and no two sources see the same thing. That would seem to be the perfect setting for meta-analysis, but not so much when it's nigh impossible to discern why differences exist.

The first issue we encountered is that the language challenge we mentioned earlier is even worse for attacks. It's almost like each source tries to use their own attack lingo and avoid that of others. As evidence of that statement, only two types of application attacks were reported by at least 3 independent sources: SQL injection (SQLi) and cross-site scripting (XSS). Among those, the prevalence of SQLi attacks ranges from 15% to 76%. XSS ranged from 4% to 54% of all observed exploits. Clearly not much in the way of consensus here, making it hard to take away more than what we already know—SQLi and XSS are common.

One topic on which all sources do agree is that web application attacks are trending up. We found historic data for 10 different reports, and every single one of them showed higher rates of attack in the most recent year. Almost seems like a conspiracy given the state of everything else, doesn't it?

We don't see this situation improving unless more sources rally around a common framework. It's clear that some have tried to do that by adopting standard vulnerability frameworks like the Open Web Application Security Project (OWASP) Top 10, but that often doesn't translate well to the attack side. For example, two sources reported attacks in the Broken Access Control category, but multiple ways exist to exploit that class of vuln. Without knowing more, defenders can't do much with that info. Which kind of defeats the purpose of reporting it, doesn't it?

Attacks Reported to F5's SIRT

Until more sources adopt more comparable ways of reporting web application attacks, the best option is leaning on credible single sources that have broad visibility across threats of interest, clearly explain their methods, and, ideally, obtain their data from environments like yours.

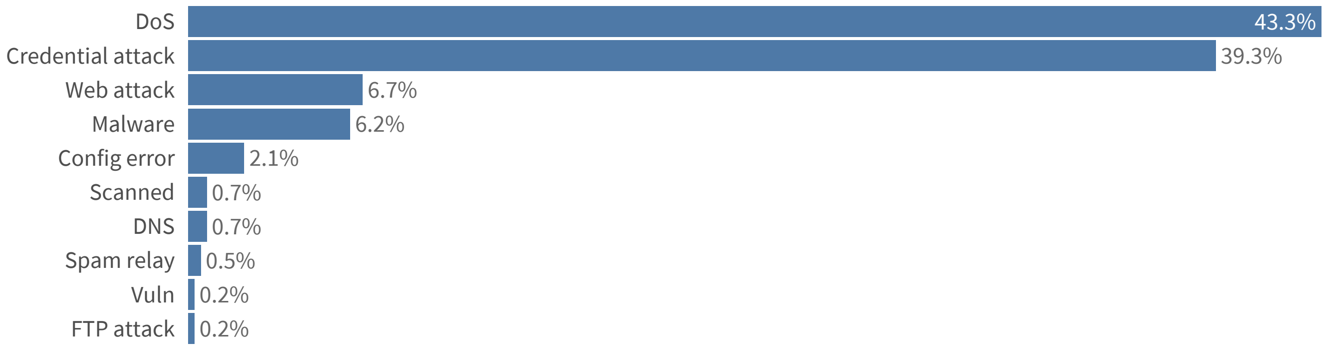


FIGURE 7: ATTACKS ESCALATED TO F5 SIRT FROM 2018 THROUGH 2020

F5, for example, has been publishing quite a bit lately about the frequency of web application exploits relative to other types of attacks they detect. They shared anonymized data with us containing attacks and incidents handled by their Security Incident Response team (F5 SIRT) from 2018 through 2020. F5 customers can escalate issues to this team, so it's a good window into threats that go beyond something that goes beyond a simple internal solution. We've categorized those events in Figure 7.

Distributed denial of service attacks (DDos) attacks take the pole position among those handled by F5 SIRT. Based on the information provided, about a quarter of these occurred at the application level. Credential attacks come next, combining brute force and credential stuffing. Both attack types involve automated attempts to log in that usually overwhelm a victim's authentication system, which is often a public-facing web application. Specific web application exploits represent 7% of the events escalated to F5 SIRT. Among those, our old friend SQLi was the most common by far.

Common Web Application Vulns

Industry reporting seems to be in a slightly better state for application vulnerabilities than for attacks, albeit still far from perfect. We gathered nine sources that present statistics on vulnerabilities discovered by their various products and services. Several of those sources reported in accordance with OWASP or CWE/SANS or they used labels we could roughly map to those categories. Overall, this was frustrating and probably not super accurate. Plus, we had to drop a good bit of information because it simply wasn't comparable.

In the end, however, we were able to shoehorn application vulnerability stats into some semblance of comparable findings based on the OWASP Top 10. Figure 8 ranks flaw categories specific to each source and tallies how often each category ranked #1 or in the top three across all sources. From that, XSS appears to be the consensus winner as the most common type of application security flaw, with two #1 votes and five top-three votes). Injection, Broken Authentication, and Sensitive Data Exposure all make a reasonable claim for runner-up.

	Contrast Labs	Veracode Vol11	Positive Tech	Veracode OpenSrc	Snyk	Acunetix*	Bulletproof*	Edgescan*
A1 Injection	7	1	5	4		3	2	1
A2 Broken Authentication	3	3	3	6	3			
A3 Sensitive Data Exposure	2	2	6	5	2			3
A4 XML External Entities (XXE)	8	5	8	7				
A5 Broken Access Control	4	8	4	3			3	
A6 Security Misconfiguration	1	6	1	8				
A7 Cross-Site Scripting (XSS)	5	4	2	1	1	1		2
A8 Insecure Deserialization	9	7		2	4			
A9 Using Components with Known Vulnerabilities			6			2	1	
A10 Insufficient Logging & Monitoring	6	9						

** These publications presented statistics on a subset, usually "critical" or "high risk"*

FIGURE 8: RANKING OF APPLICATION VULNERABILITIES REPORTED ACROSS SOURCES

You're probably wondering why we chose to rank vulnerabilities in Figure 8 rather than reporting percentages used by the original sources. We're not trying to hide anything. The honest answer is that we didn't feel the data supported that degree of comparison and that doing so would mask any semblance of consensus. By way of explanation, Figure 9 lists the minimum and maximum reported detection rates for each type of vulnerability.

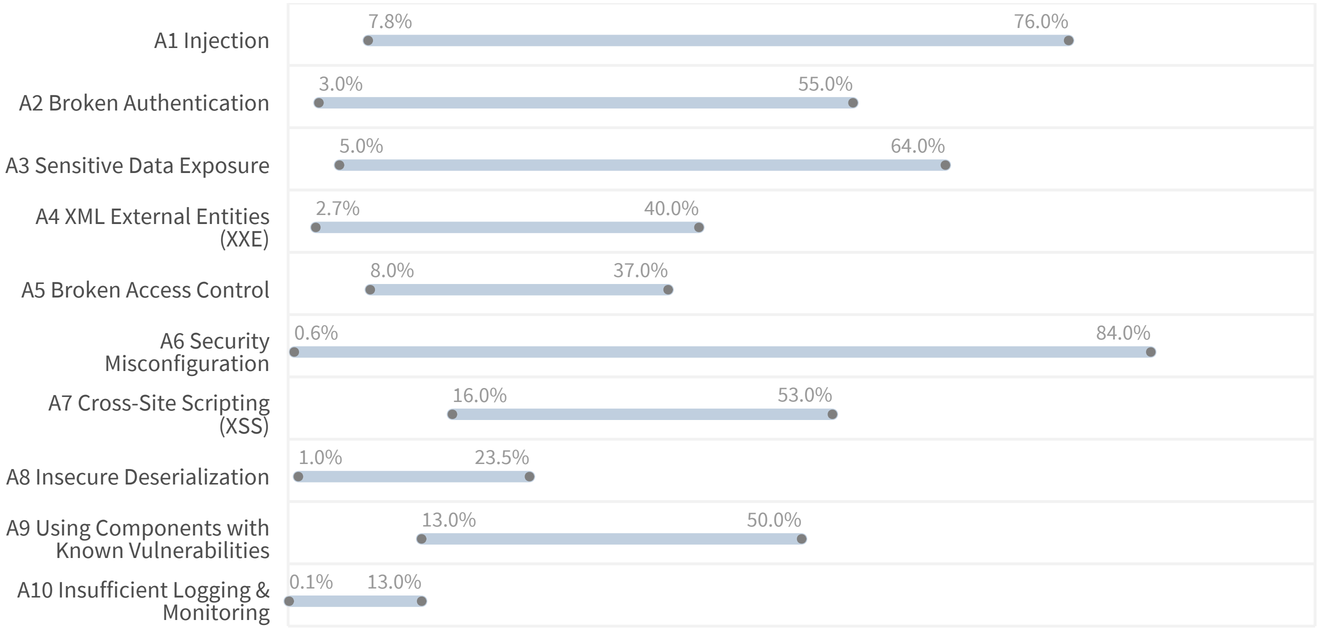


FIGURE 9: RANGE OF APPLICATION VULNERABILY PREVALENCE REPORTED ACROSS SOURCES

The min-max disparity in Figure 9 has less to do with actual disagreement about the percentage of applications with various types of flaws and more to do with what each source is measuring and how they go about it. In light of that, here's a few things to consider as you ruminate on Figures 8 and 9:

- These sources measure different types of applications (i.e., web applications vs. open-source libraries used in custom applications).
- Sources measure different stages of applications (i.e., production vs. code in active development).
- Sources measure different types of vulnerabilities (i.e., notice the asterisks in Figure 8 noting that some sources looked at all flaws, while others focused only on those with a high or critical rating).
- Sources don't all report stats according to OWASP categories (i.e., it bears repeating that we had to do some munging of the original data, introducing error).

TLDR: Attempts to analyze and compare the prevalence of various types of attacks and vulnerabilities across multiple sources suffers from a Tower of Babel effect. Lots of organizations are measuring this, but terminology and methodological differences make it hard to construct something useful from all the raw materials. Thus, the takeaways from this effort aren't all that novel or useful: There's a lot of SQLi and XSS attacks and vulnerabilities out there! If that's a disappointing outcome to you, imagine how we feel after working so hard to weave straw into gold more straw.

Concluding Thoughts

IN THEORY THERE IS NO DIFFERENCE BETWEEN THEORY AND PRACTICE, WHILE IN PRACTICE THERE IS.
- BENJAMIN BREWSTER

The idea of a report on "the state of the state of" implies that the field of security appears fragmentary and difficult to characterize. After all, if the state of the field were unambiguous, we wouldn't even be asking this kind of meta-question. Indeed, on the surface, Cyentia's conclusions uphold this fragmentary impression. The reports they analyzed, all of which come from reputable and knowledgeable sources, approach their subject matter with different definitions and assumptions. Some are focused on incidents as the most intelligible level on which to examine security. Some focus on attacker motivation, or on tactics, techniques and procedures (TTPs). Some focus on vulnerability types. This is the niche that appears the most disjointed, with the Cyentia researchers noting a "Tower of Babel" effect that prevented them from reaching conclusions more definitive than the prevalence of SQL injection and cross-site scripting.

Despite these differences, however, all these various approaches arrived at similar conclusions: attacks against web applications, most prominently authentication attacks and web exploits, constitute the greatest source of risk. Many of these reports produced similar recommendations as well, which Cyentia pithily summed up as "Fix your code; Patch your systems; Double-up your creds; Watch your back(door)." In other words, despite the panoply of tools and managed services that will do all manner of other things in the name of reducing risk, the core practices of managing risk in information systems remain the same no matter how you approach the problem. It is particularly interesting that all these recommendations are essentially about reducing attack surface and managing vulnerabilities.

The implication is that knowing what to do is not the biggest hurdle for security teams—executing it is the biggest hurdle. All CISOs probably view vulnerability management, access control, and situational awareness as critical aspects of security operations, but in practice these strategies reveal themselves as moving targets. We were surprised to see that underneath the surface, the state of the state of is not one of discontinuity and fragmentation, but one of consensus about the difficulty of execution. It appears that many security teams know what they need to do, in theory. Putting that theory into practice over time is the real problem here.

This is, in reality, quite an eye-opening conclusion. Security teams don't, in fact, need help figuring out what to do, but rather how to do it. However, creating meaningful guidance at this level of detail is challenging because all organizations are slightly different in many subtle ways—business model, organizational model, risk tolerance, technological footprint, and so on. Because of this, a shift towards a model of security intelligence and guidance that is more about how and less about what might inadvertently drive greater technological and operational conformity. Cyentia's findings indicate that this would be more of a good thing than a bad one; in the meantime, F5 Labs will turn our attention to helping drag practice closer to theory for defenders everywhere.

> “ The implication is that knowing what to do is not the biggest hurdle for security teams—executing it is. Putting theory into practice over time is the real problem.

Referenced Source Material

We reviewed upwards of 100 published reports while conducting background research for this report. As you might suspect, only a subset of those were actually used in our analysis. Below we give credit where credit is due, referencing source material used in each section.

EXTREME CYBER LOSS EVENTS

Cyentia Institute: Information Risk Insights Study 20/20 Xtreme

GENERAL INCIDENTS & BREACHES

Verizon: 2021 (and prior) Data Breach Investigations Report

Trustwave: 2020 Global Security Report

WEB APPLICATION ATT&CKS

Cisco Talos: Incident Response trends from Winter 2020-21

Crowdstrike: 2020 Global Threat Report

Cybersecurity and Infrastructure Security Agency: MITRE ATT&CK Tactics and Techniques, FY19

Infocyte: Top 20 Most Common Hacker Behaviors

Mandiant/FireEye: M-Trends 2021

McAfee Labs: Threat Report, April 2021

PICUS: The Top Ten MITRE ATT&CK Techniques

PWC: Cyber Threats 2020

Recorded Future Insikt Group: Top 6 MITRE ATT&CK Techniques Identified in 2020

Red Canary: 2021 Threat Detection Report

WEB APPLICATION ATTACKS & VULNERABILITIES

Acunetix: Web Application Vulnerability Report 2020

Bulletproof: Annual Cybersecurity Industry Report 2021

Contrast Labs: 2020 Application Security Observability Report

Edgescan: 2020 Vulnerability Statistics Report

PTSec (Positive Technologies): Web Applications Vulnerabilities and Threats: Statistics for 2019

Snyk: State of Open Source Security 2020

Veracode: State of Software Security, Volume 11

Veracode: State of Software Security, Open Source Edition