

# Why should I care about PDF application security?

## What you need to know to minimize your risk

### Table of contents

- 1: Program crashes present an opportunity for attack
- 2: Look for software that fully uses OS mitigations to decrease risk
- 3: How to evaluate a vendor's approach to security
- 4: Adobe Acrobat X Family security and independent third-party testing results
- 5: Summary

In a 2010 study across five countries in North America, EMEA, and Asia, researchers from the Ponemon Institute found that the average organizational cost of a data breach had increased to US\$4 million, up 18% from the previous year. Another study of 45 U.S.-based organizations found that cyber crime led to about one successful attack each week, at a median annual cost of US\$3.8 million per year per company and reaching as high as US\$52 million. It's getting easier for cyber criminals to target businesses, but it's not getting easier to catch or prosecute them. Today, businesses face battling cyber attackers who are intent on stealing data, crashing systems, damaging reputations, or simply showing off hacking skills.

One of the hackers' favorite attack vehicles also happens to be businesses' favorite vehicle for data, documents, and intellectual property: universally adopted file types. By embedding malicious code in files, attackers aim to gain entry to systems. PDF has become one such universally adopted file type, thanks to its status as a freely available published standard. Many developers have used the standard to create their own PDF tools, giving cyber criminals a bigger opportunity. Because hackers try to exploit weaknesses in the programs that create and view PDF files, businesses now must exercise vigilance when deciding which software to allow within their corporate environments.

This white paper discusses the consequences of a breach, the application security measures you should be looking for to prevent attacks, and the vendor characteristics that lead to secure software. It also shares the results of third-party security testing to demonstrate how Adobe® Acrobat® X and Adobe Reader® X software outperform other PDF solutions on the market when it comes to protecting organizations from attacks that could lead to disastrous consequences. It explains why businesses must assess the security of the applications they use to view and create PDF files with the same intense scrutiny they use to assess their most business-critical applications by asking, among other things:

- Which operating system (OS) mitigations are built into the software? Does the software include measures that can prevent a crash from becoming exploitable?
- What kinds of processes are in place—from product development to QA—for addressing evolving security threats?
- How does the vendor deliver security without sacrificing functionality?
- What happens after release? Does the vendor continue to actively enhance product security?
- How involved is the vendor with the broader security community?

Without this level of scrutiny, organizations open themselves to extraordinary risk.

## Program crashes present an opportunity for attack

One of the most common approaches hackers take is to feed a system corrupt files that cause it to crash, usually by fooling a document recipient into believing that a corrupt file is genuine. Crashes are disruptive and interfere with productivity, but by themselves they are not highly destructive. The bigger problem occurs when attackers try to exploit memory corruption flaws. If a crash provides an opening that attackers can successfully exploit, they can insert malicious code to run on the system. This is what enables attackers to steal data such as credit card numbers, install malware, delete files, or modify other system information on a machine that lacks adequate layered defense. This sort of attack is not at all unique to the PDF file type; hackers have used it for years against web applications, operating systems, and any common software and file type.

The impact of an attempted attack can be minimal if organizations have the right software. Otherwise, the consequences can be disastrous. Your reputation and brand can suffer irreparable damage. Your customers may lose trust and turn to your competitors. You may even be legally liable for breaches and face massive costs in court fees, punitive fines, and settlements.

Unfortunately, these worst-case scenarios are not uncommon. A simple news search on any given day will turn up numerous stories of organizations that have suffered security breaches. It might seem that attackers target only the most visible organizations, but smaller companies and government organizations are increasingly targeted, too, and hackers are targeting specific data types.

## Look for software that fully uses OS mitigations to decrease risk

Security experts widely agree that the best defense is a layered defense in depth because it protects on multiple fronts using tools built into both the application and the OS. All the following mitigations should be present in any PDF solution that you evaluate for your organization. A solution with just a few of these features does not provide the level of security that all these features combined can provide.

### Application sandbox

With a sandbox, the OS creates a confined execution environment for running programs with low rights or privileges. Sandboxes protect users' systems from being harmed by untrusted documents that might contain executable code. This method of access control works by assigning levels of integrity. A process created with low integrity is strongly limited regarding the objects that can be accessed. Other mechanisms that are often used to provide an application sandbox include restricted tokens and job object limits.

### Data Execution Prevention

Data Execution Prevention (DEP) prevents placement of data or dangerous code into memory locations that are defined as protected by the Windows® operating system. DEP provides hardware and software memory checks.

#### Non-executable memory

Hardware DEP raises an exception when code is executed from a non-executable (NX) memory location. Supported CPUs accomplish this by enabling the NX bit, marking specific memory areas as non-executable.

#### Safe structured exception handling

Software-enforced DEP checks the validity of exceptions thrown in a program to prevent malicious code from taking advantage of the exception-handling functionality. This is called safe structured exception handling (SafeSEH).

### Address space layout randomization

Even with DEP enabled, it's still possible to execute code by redirecting a function call to the address of an executable memory area in the process. To prevent such attacks, it is possible to use address space layout randomization (ASLR). This technique hides memory and page file locations of system components, making it difficult for attackers to find and target those components. Both Windows and Mac OS X v10.6 use ASLR.

### Stack cookies

The Buffer Security Check is a compiler option where a stack cookie is injected to prevent exploitation from stack-based buffer overruns. This program-wide cookie is copied between the local variables and the return address. The compiler then adds code to the prologue and epilogue of functions to stop execution if the cookie is modified.

Of the 12 tested PDF solutions, Adobe Reader X and Adobe Acrobat X are the only solutions that offer all five critical layers of security to prevent exploitation of a crash on Windows.

Productivity	Sandbox	Stack cookies	NX	ASLR	SafeSEH
Adobe Reader X	✓	✓	✓	✓	✓
Adobe Acrobat X	✓	✓	✓	✓	✓

## Other security features

Acrobat and Reader offer many other mitigation capabilities, including cross-domain protections and JavaScript whitelisting and blacklisting. Cross-domain protections specifically mitigate against cross-site scripting-based attacks, which have become more prevalent on the web. Whitelisting provides organizations with the ability to only enable JavaScript for their trusted workflows, and blacklisting protects users from attacks that target specific JavaScript API calls.

## How to evaluate a vendor's approach to security

The biggest security challenge IT teams face is that security is a moving target. New threats develop every day. That's why, when evaluating the security of PDF software, it's necessary to take into account both what the vendor builds into a product initially and what the vendor does to ensure that the software's security remains robust over time. Remaining vigilant with software security means relentlessly testing and correcting defects, while at the same time developing new protections to defend against a rapidly changing threat landscape.

### How should a vendor approach software developing and testing?

- **Security-dedicated engineering teams**—Security should be integrated at every stage of the product lifecycle.
- **Proactive security and code reviews**—Proactive incident analysis and review, as well as hardening of existing code, further drive application security improvements.
- **Participation in industry-leading security programs**—Advance sharing of product vulnerability information with security software providers, such as antivirus and intrusion detection and prevention vendors, enables the industry to work together to reduce the risk of vulnerabilities.

### How should a vendor support the updating process?

- **Predictable patching schedules**—Patching brings down IT and end-user productivity, so it should occur on a predictable and not-too-frequent schedule—for example, on the same day each month or quarter.
- **Simple configuration post-deployment**—A software solution should take advantage of OS-level tools, such as Group Policy on Windows and Property List on Mac OS, that ease the processes of modifying settings after deployment.
- **Deployment tool support**—In organizations that must update thousands of machines, support for deployment tools is critical. In particular, support for the latest software management systems, such as Microsoft System Center Configuration Manager (SCCM) and Microsoft System Center Updates Publisher (SCUP), can make it easier and more efficient to update software across the organization.

### How should a vendor support the software after release?

- **Continued enhancements**—Vendors should continue proactively working to make the software more robust and attack-proof rather than merely responding to existing threats. Occasional updates should roll out those improvements.
- **Industry collaboration**—It's important that vendors be actively involved in the security community to stay on top of the latest threats and innovations for dealing with them.

### How should a vendor respond to security bugs?

- **Transparency**—Vendors should be forthcoming about security issues and what they're doing to make their software more robust. Vendors who take security seriously proactively post and respond to threats—also known as Common Vulnerabilities and Exposures—in internationally recognized databases such as the National Vulnerability Database. A lack of vulnerability disclosures does not mean a product cannot be exploited by hackers. Rather, it might mean that the vendor who released the product is not approaching security proactively.

# Adobe Acrobat X Family security and independent third-party testing results

Engineered with security in mind, Acrobat X and Reader X incorporate industry-leading security techniques.

## Adobe outperforms other PDF solutions in security testing

In December 2011, the third-party security consulting company iSEC performed product comparison testing on 12 solutions for viewing, creating, or editing PDF documents, which it published in the report, *PDF Product Comparison*.

Testers fed each product the same set of intentionally corrupt files to attempt to induce failures. When crashes occurred, they were automatically classified as exploitable or not exploitable.

Even in the infrequent cases when testing could be made to cause a crash, not a single crash of Reader X or Acrobat X was classified as exploitable. Reader X and Acrobat X could only be made to crash 7 unique and 13 unique times, respectively. By comparison, other PDF readers crashed up to 134 unique times, and other PDF writers crashed up to 132 unique times on a large corpus of test PDF files. Testers attributed this result of zero exploitable crashes to a complete set of OS mitigations and layered defenses found in Reader X and Acrobat X, including application sandboxing, NX memory, ASLR, SafeSEH, and stack cookies. On the other hand, some tested PDF readers without these OS mitigations experienced up to 16 exploitable crashes, and some PDF writers experienced as many as 22 exploitable crashes.

In independent third-party testing, Reader X and Acrobat X experienced 0 exploitable crashes when fed corrupt files. Security features such as sandboxing prevent the possibility of exploits when a crash occurs.

Solution	Number of exploitable crashes
Adobe Reader X	0
Adobe Acrobat X	0

## Adobe invests in security and reduces out-of-band security updates

Security enhancements are part of the extensive engineering investments that Adobe has made to help harden the Acrobat product family against current and emerging threats. By continually making the software more robust against attack attempts, Adobe can help reduce or even eliminate the need for out-of-band security updates and lower the urgency of regularly scheduled updates. This helps increase operational flexibility and decreases total cost of ownership, particularly in large environments with high-security assurance requirements.

When patches do occur, Adobe's integration with leading management tools helps ensure that managed Windows desktops are always current with the latest security patches and updates, and that patches themselves are fast and simple.

## Summary

Because the PDF file type can be used for attacks, the days of licensing and deploying the lowest-cost PDF tool—without careful scrutiny of security—are long gone. When an organization's reputation and survival depend on the ability of their security defenses to stay strong in the face of repeated attacks, it's critical that they hold application vendors to a higher standard.

Adobe offers extensive mitigations to help prevent attacks from reaching their target:

- Cutting-edge sandboxing technology
- JavaScript whitelisting and blacklisting
- Disabled cross-domain access
- Streamlined patching features
- Improved tools for deployment and administration

Adobe also continues to invest in its products long after release to help make them ever more robust, so customers have the peace of mind that their security measures will continue adapting to an ever-changing threat landscape.



**Adobe**

Adobe Systems Incorporated  
345 Park Avenue  
San Jose, CA 95110-2704  
USA  
[www.adobe.com](http://www.adobe.com)

Adobe, the Adobe logo, Acrobat, and Reader are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries. Mac OS is a trademark of Apple Inc., registered in the U.S. and other countries. Windows is either a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries. All other trademarks are the property of their respective owners..

© 2011 Adobe Systems Incorporated. All rights reserved. Printed in the USA.

12/11