Campaign: How to Prepare for Penetration Tests

Shaun McCran, Technical Architect, Adobe Partner Experience

Overview

This paper describes the steps to take before initiating a penetration test against your internally hosted Campaign instance. If your Campaign instance is Adobe hosted, then collaborate with Adobe Managed Services as they manage this process. Do not instigate a penetration test on a hosted environment yourself as there is a specific process for this.

Note that there is a wealth of information out there about security in general, but this paper is specifically addressing the precursors to performing a penetration test.

Why do penetration tests?

Penetration tests are not cheap. Typically, an external company provides these services, so there is an associated fee with the task.

The penetration test results tell a powerful story to your customers. Whether it is a clean result or minor or major issues, the findings will be shared with the customer. Anything that contains a warning can make people nervous.

A bad result can cause project disruptions, as significant issues will introduce additional project tasks to remediate them. It will also affect the customer's confidence in the solution, even if they do not understand the results or what they mean. A good result has the opposite effect, as it will give the customer confidence in the solution you've just implemented.

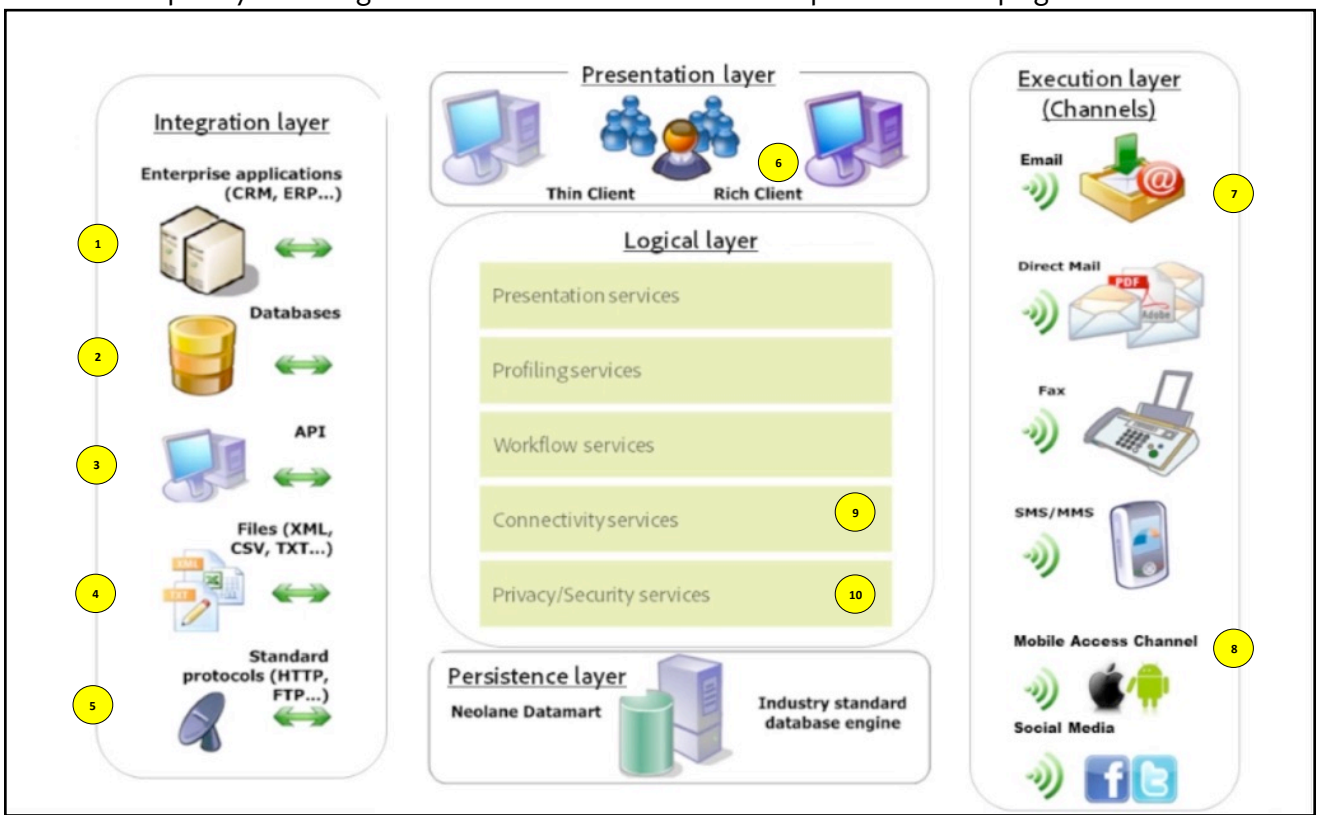
What issues might a penetration test find?

You want to deploy a secure solution. A penetration test can detail important information about the security of your solution and its surrounding infrastructure and networks. If you install something that has holes in it, it could lead to reputational damage, additional work to repair, and lead to a difficult situation with the customer regarding the future roadmap of the project.

Consider the due diligence and financial damage that not performing a penetration test might have. A breach will typically result in a leak of customer data, which comes with serious repercussions under most regional laws.

Approach to this process

Below is a frequently-used diagram that shows the architectural components of Campaign.



Item	Name	
1	Connections through to other applications	If you are connecting Campaign to existing applications in your enterprise, ensure that you are not introducing security holes via those applications. The applications you are connecting it to will have credentials that could be compromised, and network connectivity should also be secured.
2	Databases	A common way to compromise a system is to break in through the database layer, as this can have multiple weak spots. Campaign has a heavy dependency on its underlying database, whatever the technology. It is constantly reading from and writing to a database; there is a lot of traffic and a wealth of customer data there. Secure the authentication details for your database users. For example, remove the default admin accounts for database applications that install them. Lastly, like the network protocols (5), secure any ports that might be open.
3	APIs	Most APIs are secure by default and require authentication credentials to make calls against. Ensure that you have an ID layer in front of all your APIs and manage session timeouts against them or make them stateless so that there's no session at all.

4	File share locations	Campaign commonly uses files as part of its workflow process. Whether it is consuming inbound files that are put into file stores by third-party applications or generating files and storing them in outbound locations, you must ensure that these file locations are secured with credentials. Do not store them on a server that has unwarranted access across the network.
5	Network protocols	Typically, Campaign is installed into a mature network environment where DMZ zoning already exists. If not, then make sure that it is installed into a secure part of your network and only exposed via IP addresses that can communicate outwards into a public space.
6	Frontend clients	If you are allowing access to Campaign via the frontend console, then ensure that you have configured the connections over an SSL connection so that all client-server transactions are performed over a secure connection, using HTTPS.
7	Email	It's highly possible for the email channel to be abused as it is a highly performant email dispatch engine. Ensure that your email gateway connectivity is secured and transmits over SSL.
8	Mobile App integration	The Mobile SDK allows for messaging from Campaign to a user's mobile devices, via an installed application containing the SDK's APIs. Secure mobile communications services against outside interference. Refer to the API section (3) for more details.
9	Connectivity services	Make sure this is properly configured to avoid unauthorized access.
10	Security services	Make sure this is properly configured to avoid unauthorized access.

Lastly, make sure that all your software layers are up-to-date. These include:

- Operating systems: such as the server OS that is hosting Campaign.
- Database versions: there are often patches available for databases, make sure yours is patched to the latest official release.
- Campaign: Make sure to have the latest build of Adobe Campaign. Adobe regularly releases new Campaign build versions that address previous security issues or potential threats. If the build version is updated during your project, make sure to upgrade it before entering your project's test phase as a build upgrade can cause rework in testing.

Resources

Read more about Adobe's approach to security

- http://www.adobe.com/ie/security.html

Security Configuration Checklist

- http://docs.campaign.adobe.com/doc/AC/getting_started/EN/security.html

Marketing Cloud Privacy and Security Overview

- https://marketing.adobe.com/resources/help/en_US/xref/Adobe-Marketing-Cloud-Privacy-and-Security-Overview.pdf

Adobe Campaign Security Overview

- http://wwwimages.adobe.com/content/dam/acom/en/marketing-cloud/campaign/pdfs/54658.en.campaign.wp.adb-security.pdf