Implementing and Managing a Data Loss Prevention Solution

Version 1.0
Released: June 19, 2012

Securosis, L.L.C. http://securosis.com

Author's Note

The content in this report was developed independently of any sponsors. It is based on material originally posted on the Securosis blog but has been enhanced and professionally edited.

Special thanks to Chris Pepper for editing and content support.

Licensed by McAfee

McAfee, Inc., headquartered in Santa Clara, California, is the world's largest dedicated security technology company. McAfee is relentlessly committed to tackling the world's toughest security challenges. The company delivers proactive and proven solutions and services that help secure systems and networks around the world, allowing users to safely connect to the Internet, browse and shop the web more securely. Backed by an award-winning research team, McAfee creates innovative products that empower home users, businesses, the public sector and service providers by enabling them to prove compliance with regulations, protect data, prevent disruptions, identify vulnerabilities, and continuously monitor and improve their security.

http://www.mcafee.com

Copyright

This report is licensed under the Creative Commons Attribution-Noncommercial-No Derivative Works 3.0 license.

http://creativecommons.org/licenses/by-nc-nd/3.0/us/

Table of Contents

Introduction 3
- Quick Wins for long term success 3
- Getting started 4

Prepare 5
- Define incident handling process 5
- Clean directory servers 5
- Define initial priorities and process model (Quick Wins vs. Full Deployment) 6
- Map your environment 7
- Test and proof of concept 8

Integrate and Deploy Tools 9
- Define a deployment architecture 9
- Management server/appliance 12
- Directory servers 13
- Network deployment 13
- Storage deployment 15
- Endpoint deployment 16

Configure and Deploy Policies 18
- Define reports 18
- The Quick Wins process 19
- Full Deployment process 20

Manage 23
- Managing incidents 23
- Managing policies 24
- Analysis 26
- Troubleshooting 26
- Maintenance 27
- Reporting 27
- Conclusion 27

Who We Are 28
- About the Author 28

About Securosis 28

Introduction

Data Loss Prevention (DLP) is one of the farthest reaching tools in the security arsenal. A single DLP platform touches endpoints, network, email servers, web gateways, storage, directory servers, and more. There are more potential integration points than just about any other security tool — with the possible exception of SIEM. And then we need to build policies, define workflow, and implement blocking... all based on nebulous concepts like "customer data" and "intellectual property". It's no wonder many organizations are intimidated by the prospect of implementing a large DLP deployment. But based on our 2010 survey data, over 40% of organizations use some form of DLP.

Fortunately, implementing and managing DLP isn't nearly as difficult as many security professionals expect. Over the nearly 10 years we have covered the technology — talking with hundreds of DLP users — we have collected countless tips, tricks, and techniques for streamlined and effective deployments that we have compiled into straightforward processes to ease most common pain points.

We are not trying to pretend deploying DLP is simple. DLP is one of the most powerful and important tools in our modern security arsenal, and anything with that kind of versatility and wide range of integration points can be a problem if you fail to appropriately plan or test. But that's where this series steps in. We'll lay out the processes for you, including different paths to meet different needs. We will help you get up and running quickly, and stay there as efficiently and effectively as possible. We have watched the pioneers blaze the trails and hit the land mines — now it's time to share those lessons with everyone else.

Keep in mind that despite what you have heard, DLP isn't all that difficult to deploy. There are many misperceptions, in large part due to squabbling vendors (especially non-DLP vendors). But it doesn't take much to get started with DLP. On a practical note, this series is a follow-up to Understanding and Selecting a Data Loss Prevention Solution, now in its second revision. This paper picks up right where that one left off, so if you get lost in terminology you can use the other paper as a reference. Let's start with an overview and then delve into details.

Quick Wins for long term success

One of the main challenges in deploying DLP is to show immediate value without drowning yourself in data. DLP tools are generally not too bad for false positives — certainly nowhere near as bad as IDS. That said, we have seen many people deploy these tools without knowing what they wanted to look for — which can result in a lot of what we call *false real positives: real* alerts on *real* policy violations — just not things you actually care about due to the context. The way to avoid too many alerts is to deploy slowly and tune your policies, or not focus on enforcement. Here are two different implementation options:

- The Quick Wins process is best for initial deployments. Its focus is on rapid deployment and information gathering rather than enforcement, and it helps to guide full deployment later. We detailed this process in a white paper and will only briefly review it here.

- The *Full Deployment* process is for the long haul. It's a methodical series of steps to full enforcement. The goal is *enforcement* — even if that means alerting and manual response rather than automated blocking and filtering — so we spend more time tuning policies to produce useful results.

The key difference is that the Quick Wins process isn't intended to catch every single violation — just the really egregious problems. It's about getting up and running and quickly showing value by identifying key problem areas, and helping set you up for full deployment. The Full Deployment process is where you dig in, spend more time, and implement long-term policies for enforcement.

The good news is that we designed these to work together. If you start with Quick Wins all your work will feed directly into full deployment. If you already know where you want to focus, you can jump right into full deployment without bothering with Quick Wins. Either way, our process guides you around common problems and should speed up implementation.

Getting started

No matter which path you choose (Quick Wins or Full Deployment), we break the implementation process into four major steps:

1. **Prepare**: Determine which process you will use, set up incident handling procedures, prepare directory servers, define priorities, and perform some testing.
2. **Integrate and Deploy Tools**: Next you will determine your deployment architecture and integrate with your existing infrastructure. We cover most integration options — even if you only plan on a limited deployment (and no, you don't have to do everything at once).
3. **Configure and Deploy Policies**: Once the pieces are integrated you can configure initial settings and start policy deployment.
4. **Manage**: At this point you are up and running. Managing is all about handling incidents, deploying new policies, tuning and removing old ones, and system maintenance.

We struggled a bit with the language on the process since integration and deployment are closely aligned

This paper goes into depth on each step, keeping its focus on what you really need to know to get the job done. Implementing and managing DLP doesn't need to be intimidating. Yes, the tools are powerful and sophisticated, but once you know what you're doing you'll find it isn't hard to get value without killing yourself with complexity.

Prepare

One of the keys to a successful DLP deployment is preparing properly. We know that sounds a bit asinine because you can say the same thing about... well, anything... but with DLP we see a few common pitfalls in the preparation stage. Some of these steps are non-intuitive — especially for technical teams who haven't used DLP before and are focused on integration. Focusing on the following steps, before you pull the software or appliance out of the box, will significantly improve your experience. One note: clearly you need to get trained on your tool of choice before going too far, or at least read the manual. We don't list that as a separate step, but don't even think about starting your deployment without knowing how your product works. Also, while this looks like a lot, in practice you can be up and running rather quickly (a few days or weeks) once you know your priorities. Don't be scared by the granularity of reports designed to give you background information for a bunch of different deployment scenarios.

Define incident handling process

When you turn on a DLP tool you begin to collect policy violations. Most of these won't be the sort of things that require handling and escalation, but nearly every DLP deployment I have heard of quickly found things that required intervention. 'Intervention' here is a polite way of saying that someone had a talk with human resources and legal — after which that person was often escorted to the exit by a nice security gentleman in a sharp suit. It doesn't matter whether you are only doing a bit of basic information gathering or prepping a full-blown DLP deployment — it's *essential* to get your incident handling process in place before you turn on the product. I also recommend at least sketching out your processes before you get far into product selection. Many organizations involve non-IT personnel in the day-to-day handling of incidents, which affects user interface and reporting requirements. Here are some things to keep in mind:

- Criteria for escalating something from a single incident into a full investigation.
- Who is allowed access to the case and historical data — such as previous violations by the same employee — during an investigation.
- How to determine whether to escalate to the security incident response team (for external attacks) vs. to management (for insider incidents).
- The escalation workflow — who is next in the process and what their responsibilities are.
- Whether and when an employee's manager is involved. Some organizations involve line management early, while others wait until an investigation is more complete.

The goal is to have your entire process mapped out, so if you see something you need to act on immediately — especially something that could get someone fired — you have a process to manage it without causing legal headaches.

Clean directory servers

Data Loss Prevention tools tie tightly into directory servers to link incidents to users. This can be difficult because not all infrastructures are set up to tie network packets or file permissions back to a human sitting at a desk (or in a coffee

shop). Later, during the integration steps, you will tie into your directory and network infrastructure to link network packets back to users. But right now we're more focused on cleaning up the directory itself so you know which network names connect to which users, and whether groups and roles accurately reflect employees' job and rights. Some of you have completed something along these lines already for compliance reasons, but we still see many organizations with very messy directories. We wish we could say it's easy, but if you are big enough — particularly after a bunch of events like mergers and acquisitions that complicate directory infrastructures — this step may take a remarkably long time. One possible shortcut is to tie your directory to your human resources system and use HR as the authoritative source. But in the long run it's pretty much impossible to have an effective data security program without being able to tie activity to users, so you might look at something like an entitlement management tool to help clean things up.

Define initial priorities and process model (Quick Wins vs. Full Deployment)

At this point you should be in the process of cleaning your directory servers, with your incident handling process outlined in case you find anything serious early in your deployment. Now it's time to determine your initial priorities to figure out whether you want to start with the Quick Wins process or jump right into full deployment. Most organizations have at least a vague sense of their DLP priorities, but translating them into deployment priorities can be tricky. It's one thing to know you want to use DLP to comply with PCI, but quite another to know exactly *how* to accomplish that. Below is an example of mapping out high-level requirements into a prioritized deployment strategy. It isn't meant to be canonical, but should provide a good overview for most of you. Here's the reasoning behind it:

- *Compliance* priorities depend on the regulation involved. For PCI your best bet is to use DLP to scan storage for Primary Account Numbers. You can automate this process and use it to define your PCI scope and reduce assessment costs. For HIPAA the focus often starts with email to ensure that no one is sending out unencrypted patient data. The next step is often to find where that data is stored — both in departments and on workstations. If we were to add a third item it would probably be web (with webmail support), because that is a common leak vector.
- *Intellectual Property Leaks* tend to be either document based (engineering plans) or application/database based (customer lists). For documents — assuming your laptops are already encrypted — USB devices are usually a top concern, followed by webmail. You should probably also scan storage repositories, and maybe endpoints, depending on your corporate culture and the types of data you are concerned about. Email turns out to be a less common source of leaks than the other channels for IP, so it's lower on the list. If the data comes out of an application or database then we tend to worry more about network leaks (an insider or an attacker), webmail, and then storage (to figure out all the places it's stored and at risk). We also toss in USB above email, because various large leaks have shown that USB is a very easy way to move large amounts of data.
- *Customer PII* is frequently exposed by being stored where it shouldn't be, so we start with discovery again. Then — from sources such as the Verizon Data Breach Investigations Report and the Open Security Foundation DataLossDB — we know to look at webmail, endpoints and portable storage, and lastly email.

You will need to mix and match these based on your own circumstances — and we highly recommend using data-derived reports such as the ones listed above to help align your priorities with evidence, rather than operating solely on gut feel. Then adapt based on what you know about your own organization — which may include requirements such as "the CIO said we need to watch email". If you followed our guidance in Understanding and Selecting a DLP Solution you can feed the information from that worksheet into these priorities.

Now you should have a sense of what data to focus on and where to start. The next step is to pick a deployment process. Here are some suggestions for deciding which to start with. The easy answer is to almost always: start with the Quick Wins process...

- Only start Full Deployment if you have already prioritized what to protect, have a good sense of where you need to protect it, and believe you understand the scope you need to tackle. This is usually when you have a specific compliance or IP protection initiative, where the project includes well-defined data and a well-defined scope (*e.g.,* where to look for the data or monitor and/or block it).
- For everyone else we suggest starting with Quick Wins. It will highlight hot spots and help you figure out where to focus your full deployment. We'll discuss each of those processes in more depth later.

Map your environment

No matter which DLP process you select — before you can begin the actual implementation you need to map out your network, storage infrastructure, and/or endpoints. You will use this map to determine where to push out the DLP components.

1. **Network:** You don't need a complete and detailed topographical map of your network, but you do need to identify a few key components.
    1. All egress points. These are where you connect DLP monitors to SPAN or mirror ports, or install DLP inline.
    2. Email servers and MTAs (Mail Transport Agents). Most DLP tools include their own MTA, which you simply add as a hop in your mail chain, so you need to understand your chain.

   3. Web proxies/gateways. If you plan to sniff at the web gateway you will need to know where these are and how they are configured. DLP typically uses the ICAP protocol to integrate. Also, if your web proxy doesn't intercept SSL... buy a different proxy. Monitoring web traffic without SSL is nearly worthless these days.
   4. Any other proxies you might integrate with, such as instant messaging gateways.
2. **Storage:** Put together a list of all storage repositories you want to scan. The list should include the operating system type, file shares / connection types, owners, and login credentials for remote scanning. If you plan to install agents, test compatibility on test or development systems.
3. **Endpoints:** This one can be more time consuming. You need to compile a list of endpoint architectures and deployments — preferably from whatever endpoint management tool you already use for things like configuration and software updates. Mapping machine groups to user and business groups makes it easier to deploy endpoint DLP by business unit. You need system configuration information for compatibility and testing. As an example, as of this writing no DLP tool supports Macs, so you might have to rely on network DLP or exposing local file shares to monitor and scan them.

You don't need to map out every piece of every component unless you're doing your entire DLP deployment at once — which we do *not* recommend. Focus on the locations and infrastructure needed to support the project priorities established earlier.

Test and proof of concept

Many DLP users perform extensive testing or a full proof of concept during the selection process, but even if you did it's still important to push a layer deeper now that you have more detailed deployment requirements and priorities. Include the following in your testing:

- **For all architectures:** Test a variety of policies that resemble the kinds you expect to deploy, even if you start with dummy data. This is very important for testing performance — there are huge differences between using something like a regular expression to look for credit card numbers vs. database matching against hashes of 10 million real credit card numbers. And test mixtures of policies to see how your tool supports multiple policies simultaneously, and to verify which policies each component supports — for example, endpoint DLP is generally far more limited in the types and sizes of policies it supports. If you have completed directory server integration, test it to ensure policy violations tie back to real users. Finally, practice with the user interface and workflow before you start trying to investigate live incidents.
- **Network:** Integrate out-of-band and confirm your DLP tool is watching the right ports and protocols, and can keep up with traffic. Test integration — including email, web gateways, and any other proxies. Even if you plan to deploy inline (common in SMB) start by testing out-of-band.
- **Storage:** If you plan to use any agents on servers or integrated with NAS or a document management system, test them in a lab environment first for performance impact. If you will use network scanning, test for performance and network impact.
- **Endpoint:** Endpoints often require the most testing due to the diversity of configurations in most organizations, the constraints on endpoint DLP engines, and all the normal complexities of mucking with user workstations. The focus here is on performance and compatibility, along with confirming which content analysis techniques really work on endpoints (sales execs are often a bit ... obtuse ... about this). If you will use policies that change based on which network the endpoint is on, test that as well. Finally, if you are deploying multiple DLP components — such as multiple network monitors and endpoint agents — it's wise to verify they can all communicate. Some organizations find limitations here and need to adjust their architectures.

Integrate and Deploy Tools

At this point all planning should be complete. You have designed your incident handling process, started (or finished) cleaning up directory servers, defined initial data protection priorities, figured out which high-level implementation process to start with, mapped out the environment so you know where to integrate, and performed initial testing and perhaps a proof of concept. Now it's time to integrate the DLP tool into your environment. You won't be turning on any policies yet — the initial focus is on integrating the technical components and preparing to flip the switch.

Define a deployment architecture

Earlier you determined deployment priorities and mapped out your environment. Now you are ready to define the deployment architecture.

DLP component overview

We covered the DLP components a bit as we went along, but it's important to know all the technical pieces you can integrate, depending on deployment priorities. This is just a high-level overview, and we go into much more detail in Understanding and Selecting a Data Loss Prevention Solution. This list includes many possible components, but that doesn't mean you need to buy a lot of different boxes. Small and mid-sized organizations might be able to get everything except the endpoint agents on a single appliance or server.

- **Network DLP** consists of three major components and a few optional ones:
  1. **Network monitor or bridge/proxy:** This is typically an appliance or dedicated server placed inline or passively off a SPAN or mirror port. It's the core component for network monitoring.
  2. **Mail Transport Agent:** Few DLP tools integrate directly into a mail server — instead they typically insert their own MTA as a hop in the email chain.
  3. **Web gateway integration:** Many web gateways support the ICAP protocol, which DLP tools use to integrate and analyze proxy traffic. This enables more effective blocking and provides the ability to monitor SSL encrypted traffic if the gateway includes SSL intercept capabilities.
  4. **Other proxy integration:** The only other proxies we see with any frequency are for instant messaging portals, which can also be integrated with your DLP tool to support monitoring of encrypted communications and blocking before data leaves the organization.
  5. **Email server integration:** The email store is often separate from the MTA, and internal communications may never pass through the MTA — which then only has access to mail going to or coming from the Internet. Integrating directly into the message store enables monitoring of internal communications. This feature is surprisingly uncommon.
- **Storage DLP** includes four possible components:
  1. **Remote/network file scanner:** The easiest way to scan storage is to connect to a file share over the network and scan remotely. This component can be positioned close to the file repository to increase performance and reduce network saturation.

2. **Storage server agent:** Depending on the storage server, local monitoring/analysis software may be available. This reduces network overhead, runs faster, and often provides additional metadata, but may affect local performance because it uses CPU cycles on the storage server.
3. **Document management system integration or agent:** Document management systems combine file storage with an application layer and may support direct integration or the addition of a software agent on the server/appliance. This provides better performance and more context, because the DLP tool gains access to management system metadata.
4. **Database connection:** A few DLP tools support ODBC connections to scan database contents.

- **Endpoint DLP** primarily relies on software agents, although you can also scan endpoint storage using administrative file shares and the same remote scanning techniques used for file repositories. There is wide variation in the types of policies and activities which can be monitored by endpoint agents, so it's critical to understand what your tool offers.

A few other components aren't directly involved with monitoring or blocking but impact integration planning:

- **Directory server agent/connection:** Required to correlate user activity with user accounts.
- **DHCP server agent/connection:** To associate an assigned IP address with a user, which is required for accurate identification of users when observing network traffic. This must work directly with your directory server integration because the DHCP servers themselves are generally blind to user accounts.
- **SIEM connection:** While DLP tools include their own alerting and workflow engines, some organizations want to push incidents to their Security Information and Event Management tools.

Implementation priorities

It may be obvious by now, but the following charts show which DLP components and integrations with existing infrastructure you need based on your priorities.

Coverage Priority	Component	Integration Point	Notes
Network			
Network (Generic)	DLP server/appliance/sniffer	Perimeter router/switch via SPAN/mirror	Many tools unable to watch generic ports/protocols at wire speed (*e.g.,* can't watch non-standard port/protocol combinations without hard-configuring)
	DLP with embedded bridge/proxy	Inline to upstream network connection	Must support SSL interception to be useful
Email	DLP MTA	Add as hop in mail chain	Usually feature on the DLP server/appliance
	Mail server integration	Mail server	*E.g.,* direct Exchange server integration
Web (General)	DLP server/appliance/sniffer	Web gateway with SSL interception	Typically using ICAP
	DLP server/appliance/sniffer	Perimeter router/switch via SPAN/mirror	Less effective (no blocking)

Coverage Priority	Component	Integration Point	Notes
Webmail	DLP server/appliance/ sniffer	Web gateway with SSL interception	Typically using ICAP
FTP	DLP server/appliance/ sniffer	Web gateway with FTP support	
	DLP server/appliance/ sniffer	Perimeter router/switch via SPAN/mirror	Less effective (no blocking)
IM	DLP server/appliance/ sniffer	IM Proxy	Product compatibility varies widely
Storage			
File shares	DLP server/appliance	Access credentials connecting to file share	May need DLP software/ appliance installed on same subnet or switch for performance reasons
	DLP server agent software	File server	
NAS	DLP server/appliance	Access credentials connecting to file share	
	NAS Plugin	NAS server/appliance	Varies based on NAS plugin support
SAN	DLP server/appliance	Access credentials connecting to file share	For SAN exposed as a file share
	DLP server agent software	File server attached to SAN	Same agent as used for file shares
SharePoint	DLP server/appliance	Exposed WebDAV or other file shares	
	SharePoint plugin	SharePoint server	Direct plugin/integration
Other document management systems	DLP server/appliance	Exposed WebDAV or other file shares	
	Document management system plugin (product specific)	Document management system server	Varies based on document management system plugin support
Endpoint			
USB and portable media	DLP endpoint agent	Endpoint	Should support discovery, real time monitoring, and blocking

Coverage Priority	Component	Integration Point	Notes
Network traffic	DLP endpoint agent	Endpoint	Best when only used for monitoring remote devices
	DLP server/appliance/ sniffer	Perimeter router/switch via SPAN/mirror	Best as primary network monitoring tool. Endpoint agent capabilities are usually weaker
Print/Fax	DLP endpoint agent	Endpoint	
Local storage (discovery)	DLP endpoint agent	Endpoint	
	DLP server/appliance	Access credentials connecting to an administrative file share	Not recommended for performance reasons
Local storage (real time monitoring)	DLP endpoint agent	Endpoint	Supports real-time monitoring as files are saved locally vs. bulk discovery scans
Application blocking	DLP endpoint agent	Endpoint	Blocking sensitive content use in specific applications
Clipboard control	DLP endpoint agent	Endpoint	For monitoring/blocking sensitive data in cut/copy/ paste

Between this and the earlier deployment priorities chart, you should have a good idea of where to start and how to organize your DLP deployment.

Management server/appliance

With priorities fully defined, it is time to start the actual integration. The first step is set up the DLP tool itself. They come in a few flavors — and keep in mind that you often need to license different major features separately, even if they all deploy on the same box. This is the heart of your DLP deployment and needs to be in place before you can perform any integration.

- **DLP Server Software:** This is the most common option and consists of software installed on a dedicated server. Depending on your product this might actually run across multiple physical servers for different internal components (such as a back-end database) or to distribute functions for better performance. In a few cases, products require different software components running concurrently to manage different functions (such as network and endpoint monitoring). This is frequently a legacy of mergers and acquisitions — most products are converging on a single software base with, at most, additional licenses or plugins to provide additional functions. Management server overhead is usually fairly low, especially outside large enterprises, so this server often handles some amount of network monitoring, functions as the email MTA, scans at least some file servers, and manages endpoint agents. A small to medium sized organization generally only needs to deploy additional servers for load balancing, for hot standby, or to cover remote network or storage monitoring with multiple egress points or data centers. Integration is easy — install

the software and position the physical server wherever needed, based on deployment priorities and network configuration. We are still in the integration phase of deployment, and will handle the rest of the configuration later.

- **DLP Appliance:** In this scenario the DLP software comes preinstalled on dedicated hardware. Sometimes it's merely a branded server, while other appliances include specialized hardware. There is no software to install, so the initial integration is usually a matter of connecting it to the network and setting a few basic options — we will cover the full configuration later. As with a standard server, the appliance usually includes all DLP functions (which you might still need licenses to unlock). The appliance can generally run in an alternative remote monitor mode for distributed deployment.
- **DLP Virtual Appliance:** The DLP software is preinstalled into a virtual machine for deployment as a virtual server. This is similar to an appliance but requires work: to get up and running on your virtualization platform of choice, configure the network, and then set up the initial configuration options as if it were a physical server or appliance. For now just get the tool up and running so you can integrate the other components. Do not deploy any policies or turn on monitoring yet.

Directory servers

The most important deployment integration is with your directory servers (and probably DHCP servers). This is the only way to tie activity back to actual users, rather than to IP addresses. This typically involves two components:

- An agent or connection to the directory server itself to identify users.
- An agent on the DHCP server to track IP address allocation. So when a user logs onto the network their IP address is linked to their user name and then passed along to the DLP server. The DLP server can now track network activity for each user, and the directory server enables it to understand groups and roles. This integration is also required for storage or endpoint deployment. For storage the DLP tool knows which users have access to which files based on file permissions — not that they are always accurate. On endpoints the agent knows which policies to run based on who is logged in.

Network deployment

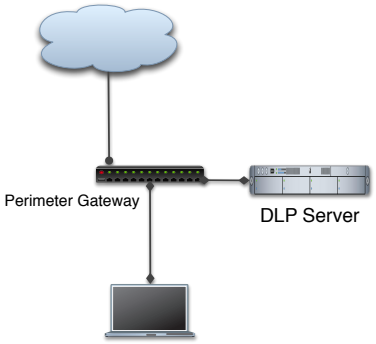
Integrating on the network is usually very straightforward — especially since much of the networking support is typically built into the DLP server. If you encounter complications they are generally:

- Due to proxy integration incompatibilities,
- Around integrating with a complex email infrastructure (such as multiple regions), or
- In highly distributed organizations with large numbers of network egress points.

Passive sniffing or bridging

Sniffing is the most basic network DLP monitoring option. There are two possible components involved:

- All full-suite DLP tools (ones that include network, storage, and endpoint features) include network monitoring capabilities on the management server or appliance. Once you install it, connect it to a network SPAN or mirror port to monitor traffic.
- The DLP server itself can normally only monitor a single network gateway (due to physical positioning), so some products also support hierarchical deployment, with dedicated network monitoring DLP servers or appliances deployed to other gateways. This might be a full DLP server with some features turned off, a DLP server for a remote location that pulls policies from and



pushes alerts back to a central management server, or a thinner appliance or application designed only to monitor traffic and send information back to the management server. Integration involves mapping network egress points and then installing appropriate hardware on the monitoring ports. High-bandwidth connections may require a server or appliance cluster; or multiple servers/appliances, each monitoring a subset of the network (either IP ranges or port/protocol ranges). If you don't have a SPAN or mirror port you'll need to add a network tap. The DLP tool needs to see all egress traffic, so a normal connection to a switch or router is inadequate.

In smaller installations you can also deploy DLP inline (bridge mode), and keep it in monitoring mode (passthrough and fail open). Even if you plan to block, we recommend that you start with passive monitoring.

Email integration

Email integrates a little differently because the SMTP protocol is asynchronous. Most DLP tools include a built-in Mail Transport Agent (MTA). To integrate email monitoring you enable the feature in the product, then add it into the chain of MTAs that route SMTP traffic out of your network. Alternatively, you might be able to integrate DLP analysis directly into your email security gateway if your vendors have a partnership. You will generally want to add your DLP tool as the next hop after your email server (e.g. Exchange server). If you also use an email security gateway that means pointing your mail server to the DLP server and the DLP server to the mail gateway. If you integrate directly with the mail gateway your DLP tool will likely add x-headers to analyzed mail messages. This extra metadata instructs the mail gateway on how to handle each message (allow, block, etc.).

Web gateways and other proxies

As we have mentioned, DLP tools are commonly integrated with web security gateways (proxies) to allow more granular management of web (and FTP) traffic. They may also integrate with instant messaging gateways, although that is very product specific. Most modern web gateways support ICAP (the Internet Content Adaptation Protocol) for extending proxy servers. If your web gateway supports ICAP you can configure it to pass traffic to your DLP server for analysis. Proxying connections enable analysis before content leaves your organization. You can, for example, allow someone to use webmail but block attachments and messages containing sensitive information. So much traffic now travels over SSL connections that you want to integrate with a web gateway that performs SSL interception (also called a "reverse proxy"). These work by installing a trusted server certificate on all your endpoints (a straightforward configuration update) and performing a 'man-in-the-middle' interception on all SSL traffic. Traffic is encrypted inside your network and from the proxy to the destination website, but the proxy has access to decrypted content.

Note: This is essentially attacking and spying on your own users, so we strongly recommend notifying them before you start intercepting SSL traffic.

If you have SSL interception up and running on your gateway, there are no additional steps beyond ICAP integration. Additional proxies, such as instant messaging, have their own integration requirements. If the products are compatible this is usually the same process as integrating a web gateway: just turn the feature on in your DLP product and point both sides at each other.

Hierarchical deployments

Until now we have mostly described fairly simple deployments, built around a single appliance or server. That's the common scenario for small and some mid-size organizations, but the rest of you have multiple network egress points to manage — possibly in very distributed situations, with limited bandwidth at each location. Hopefully you all purchased products which support hierarchical deployment. To integrate you place additional DLP servers or appliances on each

network gateway, then configure them to slave to the primary DLP server/appliance in your network core. The actual procedure varies by product, but here are some things to look out for:

- Different products have different management traffic bandwidth requirements. Some work great in all situations, but others are too bandwidth-heavy for some remote locations.
- If your remote locations don't have a VPN or private connection back to your core network, you will need to establish them for management traffic.
- If you plan to allow remote locations to manage their own DLP incidents, now is the time to set up a few test policies and a workflow to verify that your tool can support your requirements.
- If you don't have web or instant messaging proxies at remote locations, and don't filter that traffic, you obviously lose a major enforcement option. Inconsistent network security reduces DLP effectiveness — and isn't good for the rest of your security, either!
- We are only discussing multiple network deployments here, but you might use the same architecture to cover remote storage repositories or even endpoints. The remote servers or appliances receive policies pushed by your main management server and then perform all analysis and enforcement locally. Incident data is sent back to the main DLP console for handling unless you delegate to remote locations.

As we have mentioned repeatedly, if hierarchical deployment is a requirement, *please be sure to test this capability before putting money down on a product*. This is not a problem to try solving during deployment.

Storage deployment

From a technical perspective, deploying storage DLP is even easier than the most basic network DLP. You can simply point it at an open file share, load up the proper access rights, and start analyzing. The problem most people run into is figuring out which servers to target, which access rights to use, and whether the network and storage repository can handle the load.

Remote scanning

All storage DLP solutions support remotely scanning a repository by connecting to an open file share. To run they need to connect (at least administrator-only) to a share on the server to scan. But there are three common issues people encounter:

1. Sometimes it's difficult to figure out where all the servers are, and what file shares are exposed. To resolve this you can choose from a variety of network scanning tools.
2. Once you find the repositories you need access rights. And those rights need to be privileged enough to view *all* files on the server. This is a business process issue rather than a technical problem but most organizations need to do a bit of legwork to track down at least a few server owners.
3. Depending on your network architecture you may need to position DLP servers closer to the file repositories. This is very similar to a hierarchical network deployment but here the point is to reduce network impact or work around internal network restrictions — not that everyone segregates their internal networks, even though that is one of the most powerful tools in our arsenal. For very large repositories on which you don't want to install a server agent, you might need to connect the DLP server to the same switch. We have even heard of organizations adding secondary network interfaces on private network segments to support particularly intense scanning.

This is all configured in the DLP management console — that's where you configure the servers to scan, enter credentials, assign policies, and specify scan frequency and schedule.

Server agent

Server agents support higher performance without network impact, because the analysis is done right on the storage repository, with only results pushed back to the DLP server. This assumes you can install the agent, and that the server has the processing power and memory to support the analysis. Some agents also provide additional context you can't get from remote scanning. Installing the server agent is no more difficult than installing any other software, but as we have mentioned (multiple times), you need to test to understand compatibility and performance impact. Then you can configure the agent to connect to the production DLP server. Unless you run into connection issues due to network architecture, you then move over to the DLP management console to tune the configuration. The main things to set are scan frequency, policies, and performance throttles. Agents rarely run all the time — they use schedules to reduce overhead and scan during slower hours, like anti-virus software.

Depending on the product, some agents require a constant connection to the DLP server — typically because they send (compressed) data to the server for analysis rather than checking everything locally. This is very product specific, so work with your vendor to figure out which option works best for you — especially if their server agent's internal analysis capabilities are limited compared to the DLP server's. As an example, some document and database matching policies impose high memory requirements which are infeasible on a storage server, but may be acceptable on the shiny new DLP server.

Document management system/NAS integration

Certain document management systems and Network Attached Storage products expose plugin architectures or other mechanisms that allow DLP tools to connect directly rather than relying on open file shares. This method may provide additional context and information, as with a server agent. This is extremely dependent on which products you use, so we can't provide much guidance beyond "do what the manual says".

Database scanning

If your product supports database scanning you will usually make a connection to the database using an ODBC agent and then configure what to scan. As with storage DLP, deployment of database DLP may require extensive business process work: to find the servers, get permission, and obtain credentials. Once you start scanning, it is extremely unlikely you will be able to scan all database records. DLP tools tend to scan the table structure and table names to pick out high-risk areas such as credit card fields, and then scan a certain number of rows to see what kind of data is in the fields. So the process becomes:

1. Identify the target database.
2. Obtain credentials and make an ODBC connection.
3. Scan attribute names (field/column names).
4. (Optional) Define which fields to scan/monitor.
5. Analyze the first *n* rows of each identified field. We only scan a certain number of rows because the focus isn't on comprehensive realtime monitoring — that's Database Activity Monitoring — and to avoid unacceptable performance impact. But scanning a small number of rows should be enough to identify which tables hold sensitive data, which is hard to do manually.

Endpoint deployment

Endpoints are, by far, the most varied component of Data Loss Prevention. There are huge differences between the various products on the market, and far more severe performance constraints implicit in running on general-purpose workstations and laptops rather than on dedicated servers. Fortunately, as widely as the features and functions vary, the deployment process is consistent.

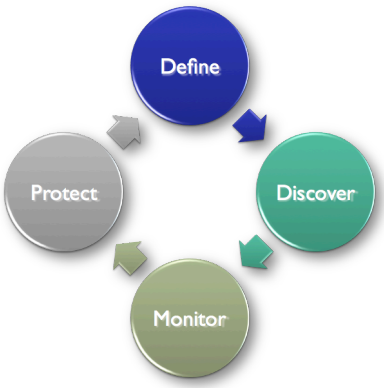
1. **Test, then test more:** I realize I have told you to test your endpoint agents at least 3 times by now, but that's not an accident — inadequate testing is the single most common problem people encounter. If you haven't already, make sure you test your agents on a variety of real-world systems in your environment to make sure performance and compatibility are acceptable.
2. **Create a deployment package or enable in your EPP tool:** The best way to deploy the DLP agent is to use whatever software distribution tool you already use for normal system updates. This means building a deployment package with the agent configured to connect to the DLP server. Remember to account for any network restrictions that could isolate endpoints from the server. In some cases the DLP agent may be integrated into your existing EPP (Endpoint Protection Platform) tool. Most often you will need to deploy an additional agent, but if it is fully integrated you can configure and enable it either through the DLP management console or in the EPP tool itself.
3. **Activate and confirm agent installation:** Once the agent is deployed go back to your DLP management console to validate that systems are covered, agents are running, and they can communicate with the DLP server. Don't turn on any policies yet — for now just confirm that the agents deployed successfully and are communicating.

Configure and Deploy Policies

Up to now we have focused on all the preparatory work before you finally flip the switch and start using your DLP tool in production. While it seems like a lot, in practice (assuming you know your priorities) you can usually be up and running with basic monitoring in a few days. With the pieces in place, it is now time to configure and deploy policies to start real monitoring and enforcement. Earlier we explained the differences between the Quick Wins and Full Deployment processes. The simplest distinction is that Quick Wins is more about information gathering and refining priorities and policies, while Full Deployment is all about enforcement. With Full Deployment you respond and investigate every incident and alert. With Quick Wins you focus more on the big picture. To review:

We generally recommend you start with Quick Wins, which gives you a lot more information before jumping into Full Deployment, and might even realign your priorities. Either way, it helps to follow the DLP Cycle. These are the four high-level phases of any DLP project:

1. **Define:** Determine the data or information you want to discover, monitor, and protect. Definition starts with a statement like "protect credit card numbers", but must be converted into a granular rule suitable for a DLP tool.
2. **Discover:** Find the information in storage or on your network. Content discovery determines where the data in question resides, while network discovery determines where it's currently being moved around on the network, and endpoint discovery is effectively content discovery but on employee computers. Depending on your project priorities, you may want to start with a surveillance project to figure out where things are and how they are being used. This phase may also entail working with business units and users to change habits before you go into full enforcement.
3. **Monitor:** Ongoing monitoring, with policy violations generating incidents for investigation. In Discover you focused on what should be allowed and setting a baseline; in Monitor your capture incidents that deviate from that baseline.
4. **Protect:** Rather than identifying and manually handling incidents, this is real-time automated enforcement — such as blocking network connections, automatically encrypting or quarantining emails, blocking files from moving to USB, or removing files from unapproved servers.



Define reports

Before you jump into deployment we suggest defining your initial report set. You'll need these to show progress, demonstrate value, and communicate with other stakeholders. Here are a few starter ideas for reports:

- Compliance reports are a no-brainer and included in many products. For example, showing you scanned all endpoints or servers for unencrypted credit card data could save significant time and resources by reducing the scope of PCI assessment.

- DLP policies are content based, so reports showing violation types by policy help figure out what data is most at risk or most in use, depending on how you design your policies. These are very useful to show management, to align your other data security controls and education efforts.
- Incidents by business unit are another great tool — even if focused on a single policy — for helping to identify hot spots.
- Trend reports are extremely valuable for showing the value of the tool and how well it helps with risk reduction. Most organizations which generate these reports achieve large reductions over time — especially when they notify employees of policy violations. Never underestimate the political value of a good report.

The Quick Wins process

We previously covered Quick Wins deployment in depth in a dedicated white paper but here is the core of the process:

The differences between a long-term DLP deployment and our "Quick Wins" approach are goals and scope. With Full Deployment we focus on comprehensive monitoring and protection of very specific data types. We know what we want to protect (at a granular level) and how we want to protect it, so we can focus on comprehensive policies with low false positives and robust workflow. Every policy violation is reviewed to determine whether it requires a response.

In the Quick Wins approach we are concerned less with incident management, and more with rapidly gaining an understanding of how information is used within the organization. There are two flavors of this approach: one where we focus on a *narrow data type*, typically as an early step toward full enforcement or to satisfy a compliance requirement, and another where we *cast a wide net* to help us understand general data usage in order to prioritize efforts. Long-term deployments and Quick Wins are not mutually exclusive — each targets a different goal and they can run concurrently or sequentially, depending on resources.

Remember: even though we aren't talking about a full enforcement process, it is essential that your incident management workflow be ready when you encounter violations that demand immediate action!

Choose Your Flavor

The first step is to decide which of two general Quick Wins approaches to take:

- **Single Type:** In some organizations the primary driver behind the DLP deployment is protection of a single data type, typically due to compliance requirements. This approach focuses only on that data type.
- **Information Usage:** This approach casts a wide net to help characterize how the organization uses information, and to identify patterns of both legitimate use and abuse. This information is often very useful for prioritizing and informing additional data security efforts.

Choose Your Deployment Architecture

Earlier you defined priorities and chose a deployment architecture, which at this point should be implemented. For the Quick Wins process you select one of the main channels (network, storage, or endpoint) rather than trying to start with all of them (this is also the way to start a Full Deployment). Network deployments typically provide the most immediate information with the least effort, but depending on what tools you have available and your organization's priorities, it may make sense to start with endpoints or storage.

Define Your Policies

The last step before hitting the 'on' button is to configure your policies to match your deployment flavor. In a single type deployment, either choose an existing category that matches the data type in your tool or quickly build your own policy. In our experience, built-in categories are available for almost all the data types that drive DLP projects. Don't worry about

tuning the policy — right now you just want to toss it out there to get as many results as possible. Yes, this is the opposite of our recommendations for a traditional focused DLP deployment. In an information usage deployment, turn on all the policies or enable promiscuous monitoring. Most DLP tools only record activity when there are policy violations, which is why you must enable the policies, although a few can monitor general activity (either full content or metadata only) without a policy trigger. Either way, the goal is to collect as much information as possible, to identify usage patterns and potential issues.

Monitor

Now it's time to turn on your tool and start collecting results. Don't be shocked — both deployment types produce much more information than a focused deployment, including more potential false positives. Remember, you aren't concerned with managing every single incident, but want a broad understanding of what's going on on your network, in endpoints, or in storage.

Analyze

Now we get to the most important part of the process — turning all that data into useful information. Once we collect enough data it's time to start analysis. Our goal is to identify broad patterns and identify any major issues. Here are some examples of what to look for:

- A business unit sending out sensitive data unprotected as part of a regularly scheduled job.
- Which data types broadly trigger the most violations.
- The volume of usage of certain content or files, which may help identify valuable assets that don't cleanly match a pre-defined policy.
- Particular users or business units with higher numbers of violations or unusual usage patterns.
- False positive patterns, for tuning long-term policies later.

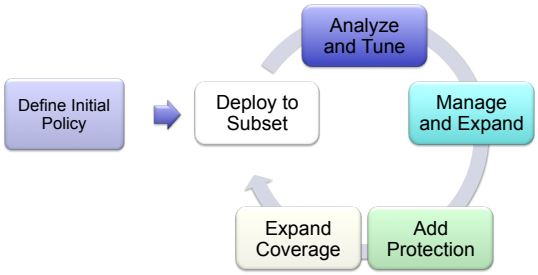
All DLP tools provide some level of reporting and analysis, but ideally your tool will allow you to set flexible criteria to support the analysis.

Full Deployment process

Even if you start with the Quick Wins process, as we recommend, you will eventually want to move into Full Deployment. This is the process you will use every time you add policies or your environment changes (such as adding endpoint monitoring to an existing network deployment).

Before we get too deep, keep in mind that we are breaking things out to an extreme level of detail, to cover the widest range of organizations and requirements. Many of you won't need to go this deep due to the nature of your policies and priorities, or your organization's size. Don't get hung up on our multi-step process — you likely won't need to move this cautiously, and may be able to run through multiple steps in a single day.

The key to success is to think incrementally — all too often we encounter organizations which turned on a new default policy and then tried to handle all the resulting incidents immediately. While DLP generally doesn't have a high rate of true false positives, that does not mean you won't get a lot of hits on a new untuned policy. For example, if you set a credit card policy incorrectly, you will alert more on employees buying sock monkeys on Amazon with their personal cards than on major



leaks. In the Full Deployment process you pick a single type of data or information to protect, create a single policy, and slowly roll it out and tune it until you reach full coverage. As with everything DLP this can move pretty quickly, or it could take months to work out the kinks in a complex policy — particularly if you are trying to protect data that's hard to distinguish from allowed usage. At a high level this follows the DLP Cycle, but we will go into greater depth.

Define the policy

This maps to your initial priorities. Start with a single kind of information, which you can identify programmatically. Examples include:

- Credit card numbers
- Account numbers from a customer database
- Engineering plans from a particular server or directory
- Healthcare data
- Corporate financials

Picking a single type to start with helps reduce management complexity and makes tuning the policy easier.

The content analysis policy itself needs to be as specific as possible, while reflecting the real-world usage of the data to be protected. For example, if you don't have a central source where engineering plans are stored, it will be hard to properly identify and protect them. You might be able to rely on a keyword, but that tends to result in too many false positives. For customer account numbers you might need to pull directly from a database if, for example, there's no pattern other than 7 or 10 digit numbers (which, if you have customers in the US, will be a problem).

We covered content analysis techniques in Understanding and Selecting a Data Loss Prevention Solution, and suggest you review it while deciding which content analysis techniques to use. The paper includes a worksheet to help guide you through the selection. In most cases your vendor will provide some pre-built policies and categories to jumpstart your own policy development. It's totally appropriate to start with one of them and evaluate its results.

Deploy to a subset

The next step is to deploy the policy in monitoring mode to a limited subset of your coverage scope. This keeps the number of alerts down and provides time to adjust the policy. For example:

- In a network deployment, limit yourself to monitoring a smaller range of IP addresses or subnets. Also start with a specific channel, such as email, before adding web or general network monitoring. If your organization is big enough, you'll subset both ways to start.
- For endpoints, limit yourself to both a subset of systems and a subset of endpoint options. Don't try to monitor USB usage, copy and paste, and local storage all at once — pick one to start.
- For storage scanning pick either a single system or even a subdirectory, depending on the volume of data involved. The key is to start small so you don't get overloaded with tuning. It's much easier to grow a smaller deployment than deal with overwhelming fallout from a poorly-tuned policy. Stick to monitoring mode so you don't accidentally break things.

Analyze and tune

You should start seeing results as soon as you turn the tool on. Hopefully you followed our advice and have your incident response process ready to go, because you will probably find things that require escalation — even without really trying.

During analysis and tuning you iteratively look at results and adjust policy. If you see too many false positives, or real positives allowed in that context, you adjust the policy. An example might be refining policies to apply differently to

different user groups (executives, accounting, HR, legal, engineering, etc.). Or you might need to toss out your approach in favor of a different option — such as switching from a pattern-based rule to database fingerprinting/matching because the targeted data is too similar to non-sensitive data in wide use.

Perhaps your tool supports full network capture, or you are using DLP in combination with a network forensics tool. If so you can collect a bunch of traffic and test policy changes against it immediately — rather than tuning a policy, running it for a few days or weeks to see results, and then tuning again. You also need to test the policy for false negatives by generating traffic (such as email messages — it doesn't need to be fancy). The goal is to align results with expectations and objectives during the limited deployment.

Manage incidents and expand scope

Once the policy is tuned you can switch into full incident-handling mode. This doesn't include preventative controls such as blocking, but instead fully investigate and handle incidents. At this point you should start generating user-visible alerts and working with business units and individual employees to change habits. Some organizations incorrectly believe it's better not to inform employees that they are being monitored or when they violate policies, so they don't attempt to circumvent security and it's easier to detect malicious activity. This is backwards — in every DLP deployment we are aware of, the vast majority of risk has been due to employee mistakes or poorly managed business processes rather than malicious activity. The evidence in the DLP deployments we have seen clearly shows that educating users *when they make mistakes* dramatically reduces the number of overall incidents.

Since user education reduces incidents so effectively, we suggest taking time to work within the limited initial deployment scope — this helps keep overhead low as you expand scope. As you get results and begin to satisfy your goals you slowly expand the scope by adding additional network, storage, or endpoint coverage.

As you expand in stages, you continue to enforce and tune the policy and handle incidents. This allows you to adapt policies to meet the needs of different business units, and avoid being overwhelmed in situations where there are many violations — at this point it's more about real violations than false positives. If you are a smaller organization or don't experience too many violations with a policy you can mostly skip this step, but we suggest starting small — even if you only spend a day on it.

Protect iteratively

At this point you will be dealing with a smaller number of incidents — if any. If you want to implement automatic enforcement, such as network filtering or USB blocking, now is a good time. Some organizations prefer to wait a year or more before moving into enforcement, and there's nothing wrong with that. But don't try to implement preventative controls with too broad an initial scope. As with monitoring, we suggest you start iteratively, allow time to deal with all the support calls, and ensure blocking is working as you expect.

Add component/channel coverage

At this point you should have a reliable policy, broadly implemented, potentially blocking policy violations. The next step is to expand by adding additional component coverage (such as adding endpoints to a network deployment) or expanding channels within a component (additional network channels like email or web gateway integration, or additional endpoint functionality). This, again, provides an opportunity to tune policies to conditions.

As we said earlier, many organizations will be able to blast through some basic policies pretty quickly without being overloaded with results. But it's still a good idea to keep this more incremental process in mind in case you need it. If you started with Quick Wins you will have a good idea of the amount of effort needed to tune your policies before you even start.

Manage

Managing DLP is generally not overly time consuming, unless you are running badly defined policies. Most of your time in the system is spent on incident handling, followed by policy management. To give you some numbers, the average organization can expect to need about one full time person (or equivalent) for every 10,000 monitored employees. This is really just a rough starting point — we have seen ratios as low as 1:25,000 and as high as 1:1,000, depending on the nature and number of policies.

Managing incidents

After deployment of the product and your initial policy set, you will likely need fewer people to manage incidents. Even as you add policies you might not need additional people, as just having a DLP tool and managing incidents improves user education and reduces the number of incidents. Here is a typical incident management process:

Manage incident handling queue

The incident handling queue is the user interface for managing incidents. This is where incident handlers start their day, and it should have certain key features:

- Ability to customize the incident for the individual handler. Some are more technical and want to see detailed IP addresses or machine names, while others focus on users and policies.
- Incidents should be pre-filtered based on the handler. In a larger organization this allows you to automatically assign incidents based on type of policy, business unit involved, and so on.
- The handler should be able to sort and filter at will; especially based on type of policy or severity of incident (usually the number of violations — *e.g.,* a million account numbers in a file, but not 5 numbers).
- Support for one-click disposition to close, assign, or escalate incidents right from the queue without having to open each one individually.

ID	Time	Policy	Channel	Severity	User	Action	Status
1138	1625	PII	Email	1.2 M	rmogull	Blocked	Open
1139	1632	HIPAA	IM	2	jsmith	Notified	Assigned
1140	1702	PII	HTTP	1	192.168.0.213	None	Closed
1141	1712	R&D/Product X	USB	4	bgates	Notified	Assigned
1142	1730	Financials	Storage	4	192.168.1.94	Encrypt	Escalated
1143	12/1/08	Source Code	Cut/Paste	12	sjobs	Confirm	Open

Most organizations distribute incident handling among a group of people with other responsibilities. Incidents are routed to a handler either automatically or manually, depending on policy and severity. Practically speaking, unless you are in a large enterprise, this could be a part-time responsibility for a single person, with additional people in other departments (such as legal and human resources) able to access the console or reports as needed for major incidents.

Initial investigation

Some incidents might be handled right from the initial incident queue — especially if automatic blocking was triggered. But due to the sensitivity of information DLP works with, many alerts require at least minimal investigation. Most DLP tools provide all the initial information you need when you drill down into a single incident. This may even include the email or file involved, with the policy violations highlighted in the text.

The job of the handler is to determine whether this is a real incident, its severity, and how to handle it. Useful information at this point includes a history of other violations — by that user and of that policy. This helps to determine if there is a larger issue or trend. Technical details help to reconstruct more of what actually happened, and this should all be available on a single screen to reduce the time and effort required to find necessary information. If the handler works for the security team, he or she can also dig into other data sources if needed, such as SIEM and firewall logs. This isn't something they should have to do often.

Initial disposition

Based on the initial investigation the handler closes the incident, assigns it to someone else, escalates to a higher authority, or marks it for further investigation.

Escalation and Case Management

Anyone who deploys DLP will eventually find incidents that require deeper investigation and escalation — and "eventually" often mean "within hours". The whole point of DLP is to find serious problems — which often require investigating your own employees. That's why we emphasize having a good incident handling process from the start — these cases can lead to someone being fired.

When you escalate, consider involving legal and human resources. Many DLP tools include case management features so you can upload supporting documentation and produce necessary reports, and track your investigative activities.

Close

The last (incredibly obvious) step is to close the incident. You need to determine a retention policy, and if your DLP tool doesn't support your retention requirements you can always output an incident report with all the salient details. As with much of what we have discussed, you will probably handle most incidents within minutes (or less) in the DLP tool, but we have detailed a common process for those times you need to dig in deeper.

Archive

Most DLP systems keep old incidents in the database, which obviously fills up over time. Periodically archiving old incidents (such as anything older than a year) is a good practice, especially since you might need to restore old records as part of a future investigation.

Managing policies

Any time you look at adding a significant new policy, you should follow the Full Deployment process described above, but there are various other day to day policy maintenance activities as well. These tend not to take a lot of time, but if you skip them for too long you might find your policy set getting stale and either not offering adequate security or causing other issues due to being out of date.

Policy distribution

If you manage multiple DLP components or regions you need to ensure new and revised policies are properly distributed and tuned for the destination environment. If you distribute policies across national boundaries this is especially important

— there might be legal considerations that require modified policies. For example, if you adjust a US-centric policy that has been adapted to other regions, you will need to update regional policies to maintain consistency. If you manage remote offices, with their own network connections, make sure policy updates are properly pushed out and consistent.

Adding policies

Brand new policies require the same effort as initial polices, except that you are more familiar with the system. So we suggest you follow the Full Deployment process again.

Policy reviews

As with anything else, today's policy may not work as well in a year, or two, or five. The last thing you want is a disastrous mess of stale but highly customized and poorly understood polices — as we often see on firewalls. Reviews should consist of:

- Periodic reviews of the entire policy set to see whether it still accurately reflects your needs, whether new policies are required, and whether older ones should be retired.
- Scheduled review and testing of individual policies to confirm that they still work as expected. When you create a new policy, create a recurring reminder to review it — at least annually. Run a few basic tests and review all violations of the policy over a given time period to get a sense of how it works. Review the assigned users and groups to make sure they still reflect the real users and business units in your organization.
- *Ad hoc* reviews when a policy produces unexpected results. Trending reports are useful for figuring this out — any large changes or deviations from the baseline are worth investigating at the policy level.
- Policy reviews during product updates — they may change how a policy works or provide new analysis or enforcement options.

Updates and tuning

Even effective policies need periodic updating and additional tuning. While you don't necessarily need to follow the entire Full Deployment process for minor updates, they should still be tested in monitoring mode before you move into any kind of automated enforcement.

Also make sure you communicate any noticeable changes to affected business units so you don't catch them by surprise. We have heard plenty of stories of someone in security flipping a new enforcement switch or changing a policy in a way that seriously impacted business operations. Maybe that's your goal, but it's always best to communicate and hash things out ahead of time.

If you find a policy seems really ineffective then it's time for a full review. For example, we know of one very large DLP user who had unacceptable levels of false positives on their account number protection because their account numbers were too similar to other numbers commonly in use in regular communications. They solved the problem (after a year or more) by switching from pattern matching to a database fingerprinting policy that checked against actual account numbers in a customer database.

Retiring policies

There are many DLP policies you might use for a limited time, such as a partial document matching policy to protect unreleased corporate financials. After the release date, there's no reason to keep the policy. We suggest you archive these policies instead of deleting them. And if your tool supports it, set expiration dates on policies... with notification so they don't silently deactivate, leaving behind security holes.

Backup and archiving

Even if you are doing full system backups, it's a good idea to periodically back up policy sets separately. Many DLP tools offer this as part of the base feature set. This allows you to migrate policies to new servers or appliances, or to recover policies when other parts of the system fail, without a full restore. We aren't saying such disasters are common — in fact we have never heard of one — but we are professional paranoids. Archiving old policies also helps if you need to review them with an old incident, as part of a new investigation or legal discovery.

Analysis

Analysis, as opposed to incident handling, focuses on big picture trends. We suggest three types of analysis:

Trend analysis

Often built into the DLP server's dashboard, this looks across many incidents to evaluate overall trends such as:

- Are incidents increasing or decreasing overall?
- Which policies are showing more or less incidents over time?
- Which business units experience more incidents?
- Are there any sudden increases in violations by a business unit, or of a policy?
- Are a certain type of incidents tied to a business process that can be changed to avoid them?

The idea is to mine your data to evaluate how risk is increasing or decreasing over time. It is often hard to notice these trends when you're deep in the muck of day to day incident handling.

Risk analysis

This is designed to show what you are missing. DLP tools only look for what you tell them to look for, and so can't catch unprotected data you don't have a policy for. A risk analysis is essentially the Quick Wins process. You turn on a series of policies with no intention of enforcing them — merely to gather information and see if there are any hot spots you should look at in more depth, or create dedicated policies for.

Effectiveness analysis

This helps assess the effectiveness of your DLP tool usage. Instead of looking at general reports, think of it as testing the tool again. Try some common scenarios to circumvent your DLP and figure out where you need to make changes.

Content discovery/classification

Discovery is the process of scanning storage for the initial identification of sensitive content, and tends to be a bit different than network or endpoint deployments. You can treat it the same — identify policy violations and respond to them — but many organizations view content discovery as a different process, often part of a larger data security or compliance project.

Content discovery projects often turn up huge amounts of policy violations, discovering sensitive files all over the place. Compounding the problem is the difficulty of identifying the file owner or business unit using the data, and why they have it. So you tend to need more analysis, at least with your first run through a server or other storage repository — to find the data, identify who uses and owns it, the business need (if any), and options for keeping the data more secure.

Troubleshooting

Outside of product-specific issues, problems people encounter tend to fall into the following categories:

- Too many false positives or negatives, which you can manage using our policy tuning and analysis recommendations.

- System components not talking to each other. For example, some DLP tools separate out endpoint and network management (often acquired as different products), and then integrate them at the user interface level. Aside from simple network routing issues, fixing communication issues between DLP components may require help from your vendor.
- Component integration with external tools such as web and email gateways may fail. Assuming you were able to get them to talk to each other previously, the culprit is usually a software update introducing an incompatibility. Unfortunately you'll need to run it down in the logs if you can't pick out the exact cause.
- New or replacement tools may not work with your existing DLP tool. These can be caused by swapping out a web gateway or using a new edge switch with different SPAN/port mirroring capabilities. Fortunately we don't hear about too many such problems with DLP tools, aside from getting the initial installation properly hooked into infrastructures, and tuning policies.

Maintenance

Maintenance overhead for DLP tools is relatively low, consisting mostly of five activities — two of which we already discussed:

- Full system backups, which you will definitely do for the central management server, and possibly also for any remote collectors or servers — depending on your tool. Some tools don't require backups of satellite installations, because you can simply swap in a new server or appliance and then reapply your configuration from the master server.
- Archiving old incidents to free up space and resources. Don't be too aggressive — you generally want a reasonable library of incidents to support future investigation.
- Archiving and backing up policies. Archiving policies means removing unused policies from the system, while backups include all policies. Keeping these separate from full system backups provides greater flexibility for restoring to new systems or migrating to additional servers.
- Health checks to ensure all system components are still talking to each other.
- Updating endpoint and server agents to the latest versions (after testing, of course).

Reporting

Ongoing reporting is an extremely important aspect of running a DLP tool. It helps show management and other stakeholders that you and your tool are providing value and managing risk. At minimum you should produce quarterly or monthly rollup reports on trends and summaries of overall activity. Ideally you'll show decreasing policy violations, but if there is an increase of some sort you can use that to get resources to investigate the root cause. You will also produce a separate set of reports for compliance. These may be on a per-project basis, tied to audit cycles, or scheduled like any other reports. For example, you might runn quarterly content discovery reports to show that you don't have any unencrypted credit card data in a storage repository, and provide these to your PCI assessor to reduce audit scope. Or runn monthly HIPAA reports for for your hospital's HIPAA compliance officer. Although you can have the DLP tool automatically generate and email reports, depending on your internal political environment you might want to review them before passing them along to outsiders, in case are any problems crop up. Also, it's never a good idea to name employees in general reports — restrict identification to incident investigations and case management summaries, with a limited audience.

Conclusion

We've gone pretty deep on implementing and managing DLP; far deeper than many of you will need to in order to get it up and running in your organization. We hope that our comprehensive approach provides all the background research you need to hit the ground up and running. Take what you need, skip the rest, and let us know how it works.

Who We Are

About the Author

Rich Mogull, Analyst and CEO

Rich has twenty years of experience in information security, physical security, and risk management. He specializes in data security, application security, emerging security technologies, and security management. Prior to founding Securosis, Rich was a Research Vice President at Gartner on the security team where he also served as research co-chair for the Gartner Security Summit. Prior to his seven years at Gartner, Rich worked as an independent consultant, web application developer, software development manager at the University of Colorado, and systems and network administrator. Rich is the Security Editor of TidBITS, a monthly columnist for Dark Reading, and a frequent contributor to publications ranging from Information Security Magazine to Macworld. He is a frequent industry speaker at events including the RSA Security Conference and DefCon, and has spoken on every continent except Antarctica (where he's happy to speak for free — assuming travel is covered).

About Securosis

Securosis, L.L.C. is an independent research and analysis firm dedicated to thought leadership, objectivity, and transparency. Our analysts have all held executive level positions and are dedicated to providing high-value, pragmatic advisory services.

We provide services in four main areas:

- Publishing and speaking: Including independent objective white papers, webcasts, and in-person presentations.
- Strategic consulting for end users: Including product selection assistance, technology and architecture strategy, education, security management evaluations, and risk assessments.
- Strategic consulting for vendors: Including market and product analysis and strategy, technology guidance, product evaluations, and merger and acquisition assessments.
- Investor consulting: Technical due diligence including product and market evaluations, available in conjunction with deep product assessments with our research partners.

Our clients range from stealth startups to some of the best known technology vendors and end users. Clients include large financial institutions, institutional investors, mid-sized enterprises, and major security vendors.

Securosis has partnered with security testing labs to provide unique product evaluations that combine in-depth technical analysis with high-level product, architecture, and market analysis.