REDUCE DATA SIZE AND CUT SIEM LICENSING COSTS

NXLog Ltd.

White Paper

The process of SIEM optimization is an ongoing effort within organizations as data reliability, performance, scaling and TCO associated issues continuously emerge. While these challenges can stem from a number of different sources, it is possible to solve them with a single solution: a focused log collection strategy. This white paper aims to help you develop such a strategy that ensures reliable log analytics and optimized performance while reducing overall SIEM costs.

The challenges of logging

> *Collecting everything is a common mistake in a logging environment.*

Although many regulations require a full copy of logs (for forensics purposes) a compliant SIEM operation can be conducted without it. One common fallacy is that by gathering and analyzing as many log messages as possible, there will be a better chance of understanding what is actually happening within the IT infrastructure, and that security personnel can react accordingly. In reality, this is as far from the truth as possible, and here is why.

Within an organization a SOC / SIEM receives on average, 5,000 to 10,000 alerts a day, yet a single security analyst can only validate around 100 per day. The main issue usually boils down to being overwhelmed with too many false positives or redundant (repetitively broadcast) alerts. Among the 10,000 events per day, only about 20-25 are actual threats. While this is a troubling number, what is much more troubling is the fact that they may get lost among all these other alerts.

How to develop a log collection strategy

- Clearly define your log collection use cases. Avoid logging all sources and forwarding them to a centralized server for processing. A good example of how to mitigate this would be to differentiate between critical and non-critical logs. In general audit logs hold far more security value than operation logs. Consider using a SIEM for higher value audit logs and a log management server for lower value events like operation logs.
- Identify your log rollover and archiving approach. Define which logs can be discarded, which logs can be archived, and for how long they need to be retained depending on factors like legal agreements, local regulations, and whether or not they contain personally identifiable information (PII).
- Consider adopting a multi-layer log management approach. This allows you to store low priority logs using more cost-effective solutions (since many solutions charge based on the retention period). Logs of higher importance can be sent directly to the SIEM.

- Understand the difference between high quality log collection and log availability. Logs that need to be available may be scattered in your infrastructure, like in cold storage. A very specific query might take hours to complete due to the sheer volume of relevant logs available. If there is a problem with low quality logs, establishing centralized log collection can help.
- Invest in a centralized log collection solution to further your strategic efforts. Such solutions offer a wide range of functionality that can simplify SIEM operations and make analytics more efficient, resulting in reduced overall SIEM costs.

The main pillars of logging

General logging requirements can be best described by the Open Web Application Security Project (OWASP) guidelines.

Context	Auditable events should be logged with sufficient context.
Retention	Events should be retained for long enough that delayed forensic analysis can be performed when necessary.
Format	Log data should be generated in a format suitable for centralized log management.
Integrity	High-value transactions should have an audit trail with controls to prevent tampering.
Monitoring	Suspicious activities should be detected and responded to promptly.
Compliance	All major regulatory standards and frameworks, including PCI DSS, HIPAA, ISO 27001, SOX, NERC and NIST CSF mandate establishing centralized logging in order to comply.

What is centralized log collection?

Centralized log collection, log aggregation, or log centralization is the process of sending event log data to a dedicated server or service for storage and optionally search and analytics. Storing logs on a centralized system provides several benefits versus storing the data locally.

- Event data can be accessed even if the originating server is offline, compromised, or decommissioned.
- Data can be analyzed and correlated across more than one system.
- It is more difficult for malicious actors to remove evidence from logs that have already been forwarded.

- Incident investigation and auditing is easier, as all event data is collected in one location.
- Scalable, high-availability, and redundancy solutions are easier to implement and maintain because they can be implemented at the point of the collection server.
- Compliance with internal and external standards for log data retention only need to be managed at a single point.

How can centralized log collection provide better SIEM optimization?

Reducing the number of logs achieved through log event filtering, selective event collection, classification, correlation, and removing duplicate logs.

Reducing the size of the log data by parsing out fields containing the same content or fields that are not essential for the SIEM, and by truncating long fields if the SIEM needs only a subset of the data.

Providing data compression via network and batch compression.

Providing a lightweight architecture to reduce memory footprint and increased flexibility for different uses.

Offering better storage space management by forwarding logs to multiple endpoints combined with resource splitting to manage which logs should actually go to the SIEM.

Offering better resource efficiency by managing log enrichment and log conversion to determine how and when they should be processed during the log collection cycle.

Cutting back on SIEM costs by significantly reducing the event per second (EPS) and Gigabytes per day (GB/day) log volume.

Many of our customers turn to us to meet their SIEM optimization and cost saving challenges by using NXLog as their centralized log collection solution.

Log collection approaches solved with NXLog

The how-to approach for log collection to reduce data storage and licensing costs.

Approach – reducing the number of logs

Filter log events: Depending on the logging requirements and the log source, it is possible to simply discard certain events. NXLog can filter events based on nearly any set of criteria. Read more in the

Reducing Bandwidth and Data Size section.

Typical example where dropping certain logs with a duplicated error message is needed

```
2018-10-26 09:48:26 ERROR Service is already running↵
2018-10-26 10:10:44 ERROR Service is already running↵
2018-10-26 13:22:27 ERROR Service is already running↵
2018-10-26 16:35:07 ERROR Service is already running↵
2018-10-26 16:36:45 WARNING stopping nxlog service↵
```

Selective event collection: NXLog can collect certain Windows Event IDs and drop the rest. Important security events such as lateral movements can be configured instead of collecting all event logs. Read more in the Event IDs to Monitor section.

Using structured logging

When logs are more human-readable, and security experts are spending less time having to look for a "needle in a haystack," it can dramatically reduce the operational cost of the SIEM. NXLog provides structured log collection which allows events to be classified and correlated using any of the values provided by the event set. Read more in the Using Structured Logging for Effective Log Management white paper.

Raw log sample, difficult to correlate:

```
<38>Nov 22 10:30:12 myhost sshd[8459]: Failed password for invalid user linda from
192.168.1.60 port 38176 ssh2↵
```

An excerpt of the same event as structured data, easier to correlate

```
{  "SyslogFacility": "USER",  "SyslogSeverity": "NOTICE",  "EventTime": "2019-11-22
10:30:12",  "Hostname": "myhost",  "SourceName": "sshd",  "ProcessID": 8459,  "Status":
"failed",  "AuthenticationMethod": "password",  "Reason": "invalid user",  "User":
"linda",  "SourceIPAddress": "192.168.1.60",  "SourcePort": 38176,  "Protocol": "ssh2"}
```

Approach – reducing the size of log data

Trimming log events: NXLog can trim events by removing parts of redundant text from the event record or state which fields to discard. This helps reduce log data bloat. Read more in the Trimming Events section.

Exec example of truncating a Windows Event Log event:

```
if ($Channel == 'Security') and ($EventID == 4688)
$Message =~ s/\s*Token Elevation Type indicates the type of .*$//s;
else if $(Channel == 'Security') and ($EventID == 4769)
$Message =~ s/\s*This event is generated every time access is .*$//s;
```

Deduplicate metadata: Some log data will have duplicated metadata across multiple events of the same time. One example of this is the "descriptive event data" in Windows Event Log. By removing this verbose text from common events, event sizes can be reduced significantly while still preserving the important metadata.

Approach – use data compression over the network

SSL and HTTPS compression modules: NXLog allows you to enable data compression when sending data over the network.

Batch compression: NXLog Enterprise Edition can transfer events in compressed and encrypted batches to save bandwidth. This is especially useful for transferring logs in low-bandwidth environments or on networks with suboptimal performance. Read more about Compressing During Transport.

Approach – use lightweight deployment solutions

Lightweight architecture: NXLog's agent provides a reduced memory footprint and increased flexibility for different uses. Or consider deploying an agent- less or mixed approach. Keep in mind that only agent-based solutions can provide advanced features like (most of the) trimming, filtering, and compressing. Use the same agent for your configurations instead of keeping track of multiple agents. If you are interested in learning more, read our blog: Agent-based versus agent-less log collection - which option is best?

Approach – saving storage

Log forwarding: NXLog allows you to forward logs form a single host to multiple endpoints. This mitigates the risk of log tampering or removal following an attack.

Resource splitting: NXLog can centralize logs by forwarding them to a log management server and select which sources go to the SIEM.

On-disk compression: Since NXLog can be configured to run external programs or scripts, it can automate the compression of old log files to reduce disk usage.

Approach – decide when, how, and where log enrichment takes place

Log enrichment: With NXLog you can decide how and at what point Log enrichment happens during the log collection cycle. For example, determine which logs should be enriched on the agent-side at the point of forwarding or at the point of rewriting to disk.

Resource efficiency: Converting certain data formats from one to another can be resource intensive.

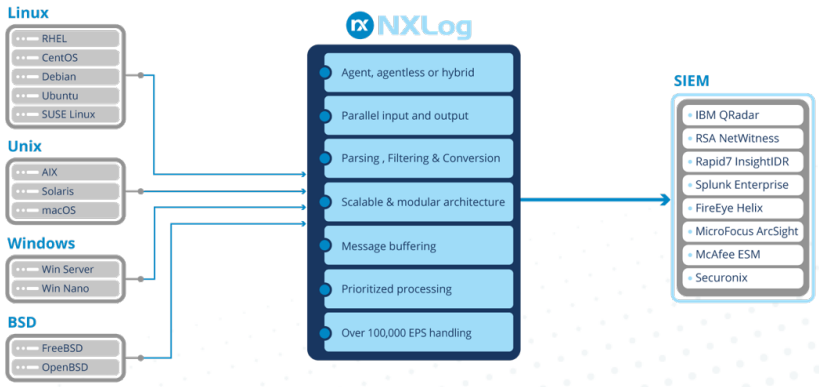
NXLog forwards and converts log data on the fly, without any need to write the converted data to local disk before forwarding, thus maintaining high efficiency and conserving resources.

The goal – cost reduction

Lower SIEM TCO costs: Generally, SIEM on the market either follow an Event per second (EPS) or Gigabytes per day (GB/day) pricing structure. In both cases the core principle is the same. The more recklessly you collect and analyze data the more expensive it becomes. Lowering the total cost of ownership of your SIEM infrastructure is an added benefit when you use NXLog to optimize log collection specifically for your use cases.

> *With NXLog, there is an 80.9% data reduction between two of the same events. See the Rapid7 InsightIDR SIEM use case and license comparison table below.*

NXLog infrastructure



NXLog benefits

NXLog provides better SIEM by simplifying the log data and reducing its size while retaining important metadata. The end result is faster search times.

NXLog increases confidence in SIEM analysis by making sure that captured logs remain tamperproof.

NXLog is a vendor neutral solution that allows you to connect and combine solutions that best fit your needs.

NXLog can be deployed across a wide variety of nodes and destinations, such as other SIEMs or other Log Management solutions, either on-premise or in the cloud.

The table below summarizes how SIEM licensing costs are determined vs the role NXLog plays in making a difference to reduce such costs.

Table 1. SIEM Licenses and the NXLog Difference

Company	Licensing Issue	What can NXLog do
Splunk	*Annual subscription pricing is based on GB/day. See calculator. Splunk free licenses have limitations stipulated in the end user agreement.*	*No endpoint restrictions or limitations. Modules are available to help decrease log data size.*
MicroFocus ArcSight	*ArcSight Logger and ArcSight Data Platform (ADP) appliances and software are licensed based on GB per day, data ingested, and security events correlated per second. See ArcSight license and enforcement information.*	*No licensing limitations or subscriptions involving log data usage. Used to pre-process logs (filtering, deduplicating metadata, dropping fields) prior to forwarding to ArcSight.*
Elastic	*Pricing depending on deployment type (cloud vs on-premise), Elastic Stack subscriptions, etc. SIEM pricing starts at the Standard license level. Even Elastic Stack security features require a standard, paid license.*	*Security features like SSL are available as default. All modules are available on the agent. No additional installation required as is the case with Beats.*
IBM QRadar	*The SIEM relies on a subscription model and a number of factors unrelated to data usage also affect the final cost. Related to data, the license costs depend on the number of events and number of flows.*	*No licensing limitations or subscriptions for log data. Used to filter, deduplicate metadata, drop fields, and take control of log data flows.*
LogRhythm	*LogRhythm has an unlimited data plan.*	*Conduct agent-side filtering to reduce EPS. The compression modules reduce bandwidth needed to send logs from remote sites.*

Use Case: Rapid7 InsightIDR SIEM

Windows account login attempts generate events containing copious metadata. See below for an authentication event in Snare Syslog format.

Full event sample of a Windows Failed Authentication Event in Syslog


```
01 Aug 2019 17:46:45.291{↵
  "timestamp": "2019-08-01T21:46:43.000Z",↵
  "hostname": "NXLOG-AGENT",↵
  "event_code": "4625",↵
  "description": "An account failed to log on.",↵
  "subject_user_sid": "S-1-0-0",↵
  "subject_user_name": "-",↵
  "subject_domain_name": "-",↵
  "subject_logon_id": "0x0",↵
  "logon_type": "Network",↵
  "target_user_sid": "S-1-0-0",↵
  "target_user_name": "ADMINISTRATOR",↵
  "target_domain_name": "",↵
  "failure_reason": "Unknown user name or bad password.",↵
  "status": "username or password incorrect",↵
  "sub_status": "user name is correct but the password is wrong",↵
  "process_id": "0x0",↵
  "process_name": "-",↵
  "workstation_name": "-",↵
  "ip_address": "212.92.116.56",↵
  "ip_port": "0",↵
  "logon_process_name": "NtLmSsp",↵
  "authentication_package_name": "NTLM",↵
  "transmitted_services": "-",↵
  "lm_package_name": "-",↵
  "key_length": "0",↵
  "source_data": "<11>Aug  1 17:46:43 NXLOG-AGENT MSWinEventLog\t3\tSecurity\t77\tThu
Aug 01 17:46:43 2019\t4625\tMicrosoft-Windows-Security-Auditing\tN/A\tN/A\tFailure
Audit\tNXLOG-AGENT\tLogon\t\tAn account failed to log on.    Subject:    Security ID:
S-1-0-0    Account Name:    -    Account Domain:    -    Logon ID:    0x0    Logon Type:    3
Account For Which Logon Failed:    Security ID:    S-1-0-0    Account Name:
ADMINISTRATOR    Account Domain:      Failure Information:    Failure Reason:    Unknown
user name or bad password.    Status:    0xC000006D    Sub Status:    0xC000006A
Process Information:    Caller Process ID: 0x0    Caller Process Name: -    Network
Information:    Workstation Name: -    Source Network Address: 212.92.116.56    Source
Port:    0    Detailed Authentication Information:    Logon Process:    NtLmSsp
Authentication Package: NTLM    Transited Services: -    Package Name (NTLM only): -
Key Length:    0    This event is generated when a logon request fails. It is generated
on the computer where access was attempted.    The Subject fields indicate the
account on the local system which requested the logon. This is most commonly a
service such as the Server service, or a local process such as Winlogon.exe or
Services.exe.    The Logon Type field indicates the kind of logon that was requested.
The most common types are 2 (interactive) and 3 (network).    The Process Information
```

```
fields indicate which account and process on the system requested the logon.    The
Network Information fields indicate where a remote logon request originated.
Workstation name is not always available and may be left blank in some cases.    The
authentication information fields provide detailed information about this specific
logon request.   - Transited services indicate which intermediate services have
participated in this logon request.   - Package name indicates which sub-protocol was
used among the NTLM protocols.   - Key length indicates the length of the generated
session key. This will be 0 if no session key was requested.\t223050"↵
}↵
```


However, only important ingress authentication details are needed. The NXLog xm_rewrite module, xm_json JSON module and drop_fields(); procedure can decrease the data size. They will also structure the data, leaving only the important event details required by the Rapid7 Insight IDR UEF.

Windows Failed Authentication Event after log enrichment


```
{"timestamp":"2019-08-01T20:33:41.000Z","user":"NXLOG-AGENT","account":"NXLOG-AGENT"
,"result":"FAILED_OTHER","source_ip":"212.92.117.25","service":"CUSTOM UNIVERSAL
EVENT","geoip_organization":"NFOrce Entertainment B.V.","geoip_country_code":"NL"
,"geoip_country_name":"Netherlands","geoip_city":"","geoip_region":"","authentication
_target":"-","source_json":{"time":"2019-08-01T20:33:41Z","account":"NXLOG-AGENT"
,"version":"v1","authentication_target":"-","source_ip":"212.92.117.25","event_type"
:"INGRESS_AUTHENTICATION","authentication_result":"FAILURE"}}
```


The difference in size between the two is 2,360 characters (2,917 vs only 557 characters), which is a reduction of 80.9%.

In conclusion

To summarize, SIEM is a technology that combines logs via data (log) correlation for supporting security analysts who oversee tons of information generated by virtually any kind of network device or computer.

There is a common misconception that collecting every single log for the SIEM will result in better analytics, but in reality all it does is increase SIEM costs and generate a large number of false alerts. The solution is the proper implementation and utilization of a log collection tool combined with SIEM analytics. The result is better log collection, reduced log noise, reliable analytics, and reduced SIEM costs.

Deploying a proper SIEM solution is a huge challenge for any IT team. And, a good SIEM system requires an even better log collection environment. Following the best practices of the industry and fine-tuning your security frequently, leads to better IT Security operations.

About NXLog

NXLog provides a reliable centralized log collection across the entire enterprise. For more information on NXLog visit our website or schedule a meeting with our professionals.

NXLog Ltd. develops multi-platform log collection tools that support many different log sources, formats, transports, and integrations. The tools help administrators collect, parse, and forward logs so they can more easily respond to security issues, investigate operational problems, and analyze event data. NXLog distributes the free and open source NXLog Community Edition and offers additional features and support with the NXLog Enterprise Edition.