Edward Seidl – eseidl@us.ibm.com
Chris Meyer, CISSP – meyerchr@us.ibm.com

October 2023

IBM z/OS Communications Server

z/OS Encryption Readiness Technology (zERT) Overview

Background: Cryptographic network protection on z/OS

z/OS provides 4* main mechanisms to protect TCP/IP traffic:

1 **TLS/SSL direct usage**

- Application is explicitly coded to use these
- Configuration and auditing is unique to each application
- Per-session protection
- TCP only

2 **Application Transparent TLS (AT-TLS)**

- TLS/SSL applied in TCP layer as defined by policy
- Configured in AT-TLS policy via Configuration Assistant
- Auditing through SMF 119 records
- Typically transparent to application
- TCP/IP stack is user of System SSL services

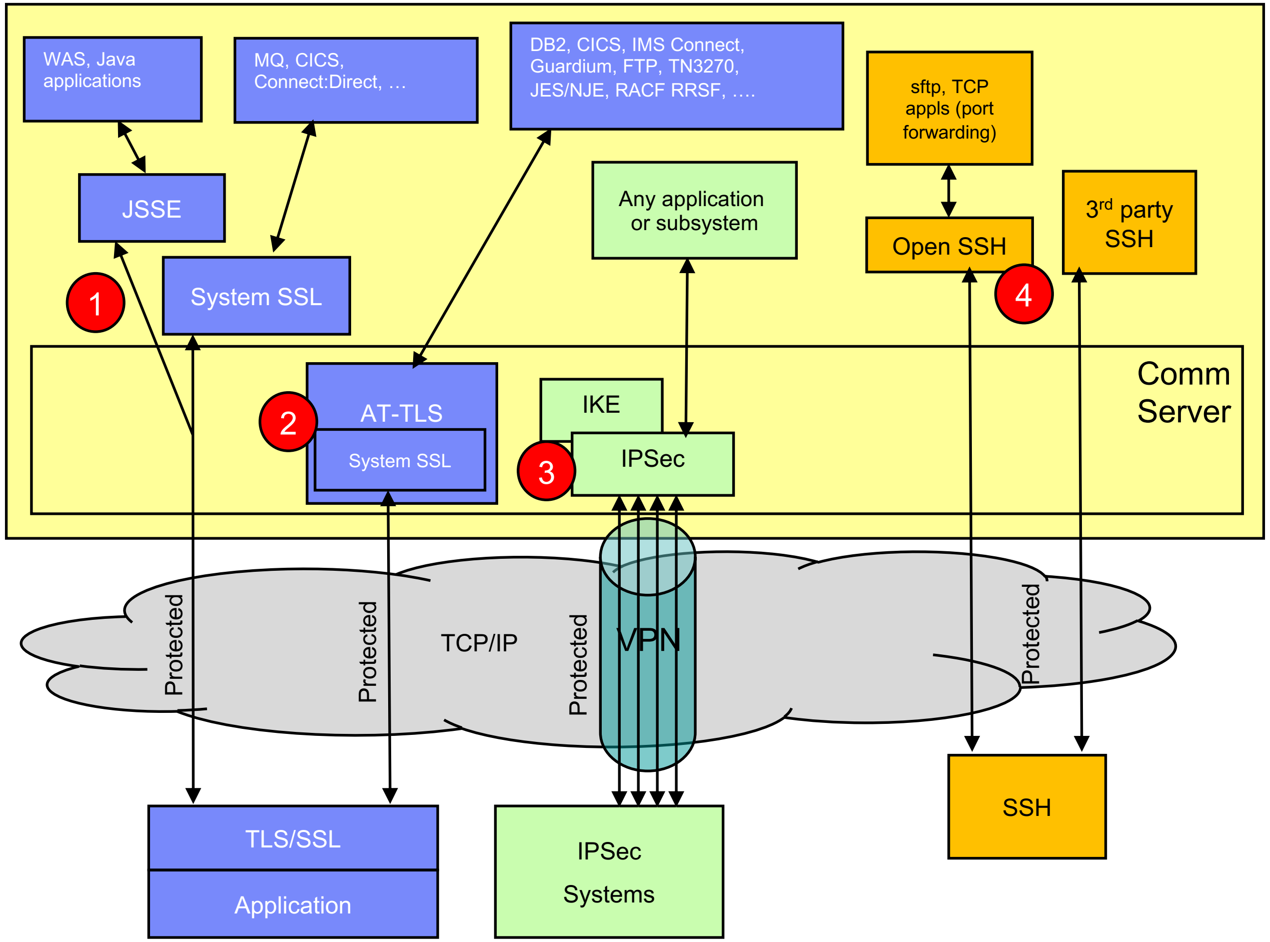
3 **Virtual Private Networks using IPSec and IKE**

- “Platform to platform” encryption
- IPSec implemented in IP layer as defined by policy
- Auditing through SMF 119 records – tunnel level only
- Completely transparent to application
- Wide variety (any to all) of traffic is protected
- Various topologies supported (host to host, host to gateway, etc.)
- IKE negotiates IPSec tunnels dynamically

4 **Secure Shell using z/OS OpenSSH**

- Mainly used for sftp on z/OS, but also offers secure terminal access and TCP port forwarding
- Configured in ssh configuration file and on command line
- Auditing via SMF 119 records
- TCP only

(Can also have 3rd party SSH implementations)



* - z/OS also provides Kerberos support, but that is not covered in this presentation

z/OS Encryption Readiness Technology (zERT) overview

- With all this complexity, how can you tell…

Which traffic is being protected? Which is not?

How is the traffic being protected?

Who does the traffic belong to?

Do existing and new configurations adhere to your company's security policies?

- zERT is design specifically to answer the above questions
  - Positions the **TCP/IP stack** as a central collection point of cryptographic protection attributes for:
    - **TCP** connections that are protected by **TLS, SSL, SSH, IPsec** or are **unprotected***
    - **Enterprise Extender** connections that are protected by **IPsec** or are **unprotected***
  - Two methods for discovering the security sessions and their attributes:
    - **Stream observation** (for TLS, SSL and SSH) – the TCP/IP stack observes the protocol handshakes as they flow over the TCP connection
    - **Advisory observation** by the cryptographic protocol provider (System SSL, ZERTJSSE provider, z/OS OpenSSH, and z/OS IPsec are enabled for zERT advisory observation)
  - Reported through SMF 119 records via:
    - **SMF** and/or
    - **Real-time** network management interfaces (NMIs)

unprotected* = no protection that zERT recognizes

What does zERT collect, record and monitor?

- **Significant attributes**
  - Identifying attributes like IP addresses, ports, jobname, userid, etc. (subtype 11 and 12)
  - Protection attributes like protocol version, cryptographic algorithms, key lengths, etc. Changes in these cause a protection state change record to be written if they change (subtype 11 and 12)

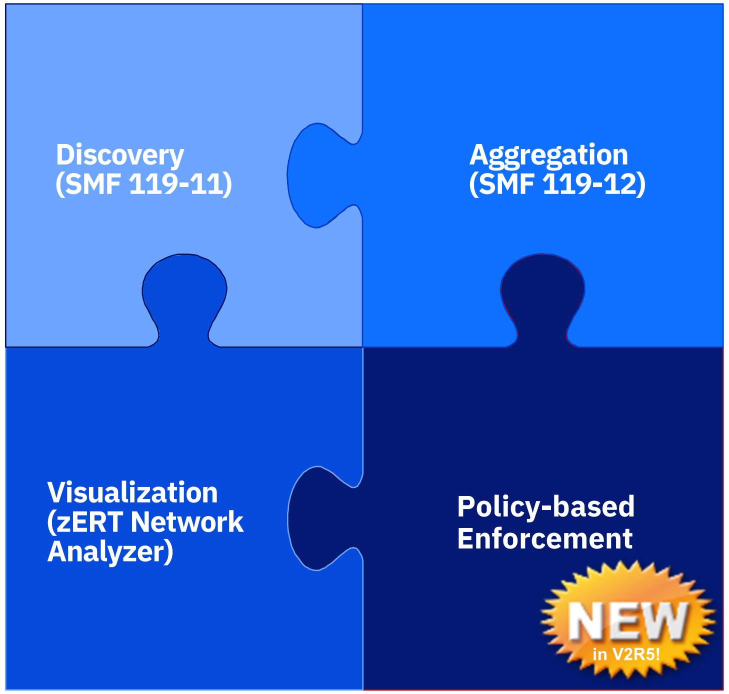
- **Informational attributes** – protection attributes like protocol session identifiers, session or certificate expiry data and certificate serial numbers are recorded for informational purposes only. Changes in these attributes do not affect the strength of the cryptographic protection (subtype 11 only)

- ***zERT does not collect, store or record the values of secret keys, initialization vectors, or any other secret values that are negotiated or derived during cryptographic protocol handshakes***

See the z/OS Communications Server IP Programmer's Guide for all the details

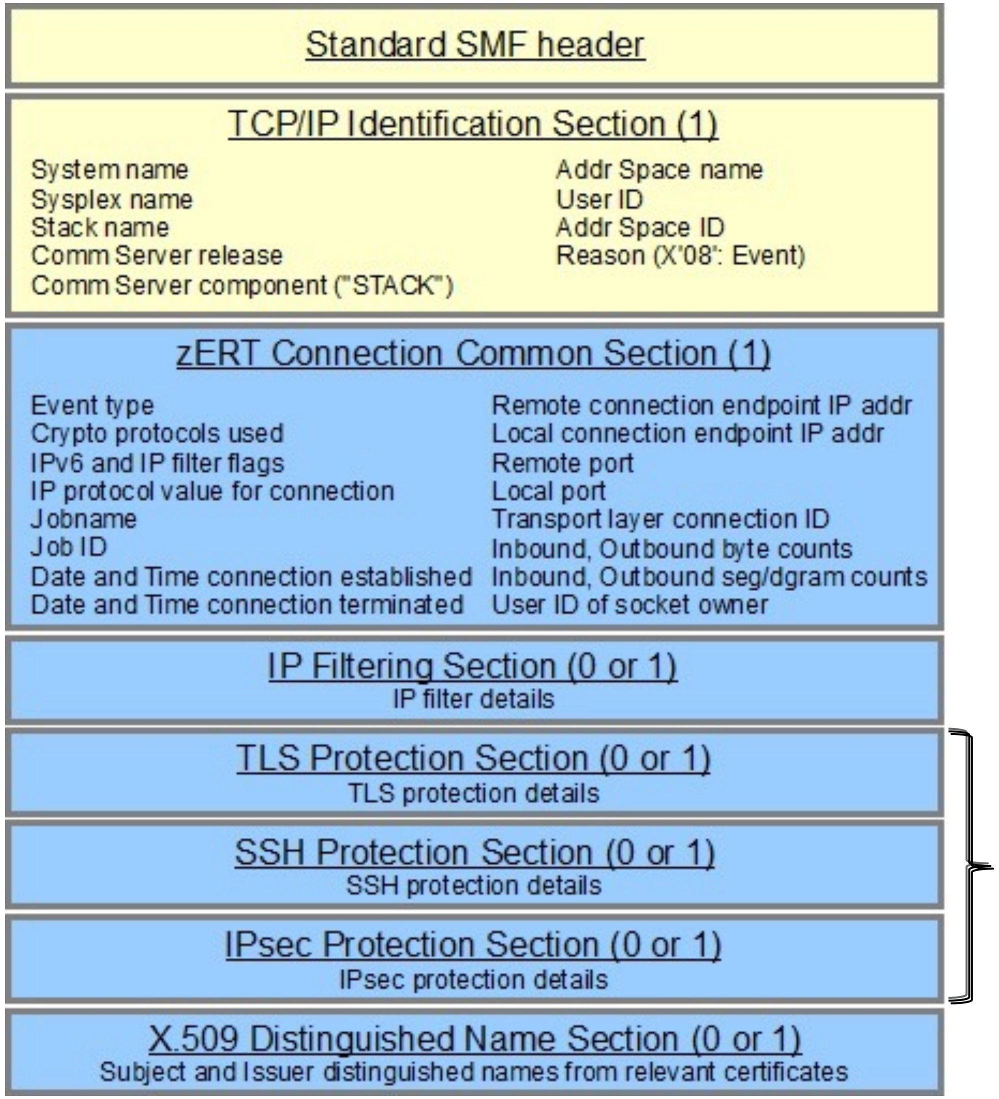
zERT features

- zERT **Discovery**
  - **SMF 119 subtype 11 “zERT Connection Detail” records**
  - These records **describe the complete cryptographic protection history of each TCP and EE connection**
  - **At least one record** is written **for each connection** - and each describes **all cryptographic protection** for that connection
  - Well suited for **real-time monitoring** applications
  - Depending on your z/OS network traffic, these could be generated in very high volumes
- zERT **Aggregation**
  - **SMF 119 subtype 12 “zERT Summary” records**
  - These records **describe the repeated use of security sessions over time**
  - Writes **one zERT Summary record at the end of each recording interval for each security session** active during the interval
  - Well suited for **reporting and analysis**
  - Can greatly reduce the volume of SMF records (over Discovery) while providing the same level of cryptographic detail
- zERT **Network Analyzer**
  - **Web-based (z/OSMF) UI** to query and analyze zERT Summary (subtype 12) records
  - **The latest network analyzer PTF always contains an up-to-date fresh install image**
  - Intended for z/OS network security administrators (typically systems programmers)
  - Comes with Communications Server at **no extra charge, but relies on Db2 for z/OS 11 or 12**
- zERT **Policy-based Enforcement – new in z/OS V2R5**
  - **Real-time monitoring based on user-written policy rules**
  - Provides **notification or even defensive actions** when insufficient cryptographic protection is recognized



zERT Discovery: SMF Connection Detail record (type 119, subtype 11)

- **At least one Connection Detail record is written for each TCP or EE connection**. Written at various events in a TCP or EE connection’s life.
- **Describes all of the cryptographic protection applied to a specific connection, including significant changes** to cryptographic protection during the life of the connection
- Examples
  - A record for a connection protected by a TLS session and an underlying IPsec tunnel will contain both a TLS protection section and an IPsec protection section
  - A record for a connection with no recognizable protection will have no protocol-specific sections
  - A record for a connection protected by TLS alone will have a TLS protection section.



Standard SMF header

TCP/IP Identification Section (1)

System name
Sysplex name
Stack name
Comm Server release
Comm Server component ("STACK")

Addr Space name
User ID
Addr Space ID
Reason (X'08': Event)

zERT Connection Common Section (1)

Event type
Crypto protocols used
IPv6 and IP filter flags
IP protocol value for connection
Jobname
Job ID
Date and Time connection established
Date and Time connection terminated

Remote connection endpoint IP addr
Local connection endpoint IP addr
Remote port
Local port
Transport layer connection ID
Inbound, Outbound byte counts
Inbound, Outbound seg/dgram counts
User ID of socket owner

IP Filtering Section (0 or 1)
IP filter details

TLS Protection Section (0 or 1)
TLS protection details

SSH Protection Section (0 or 1)
SSH protection details

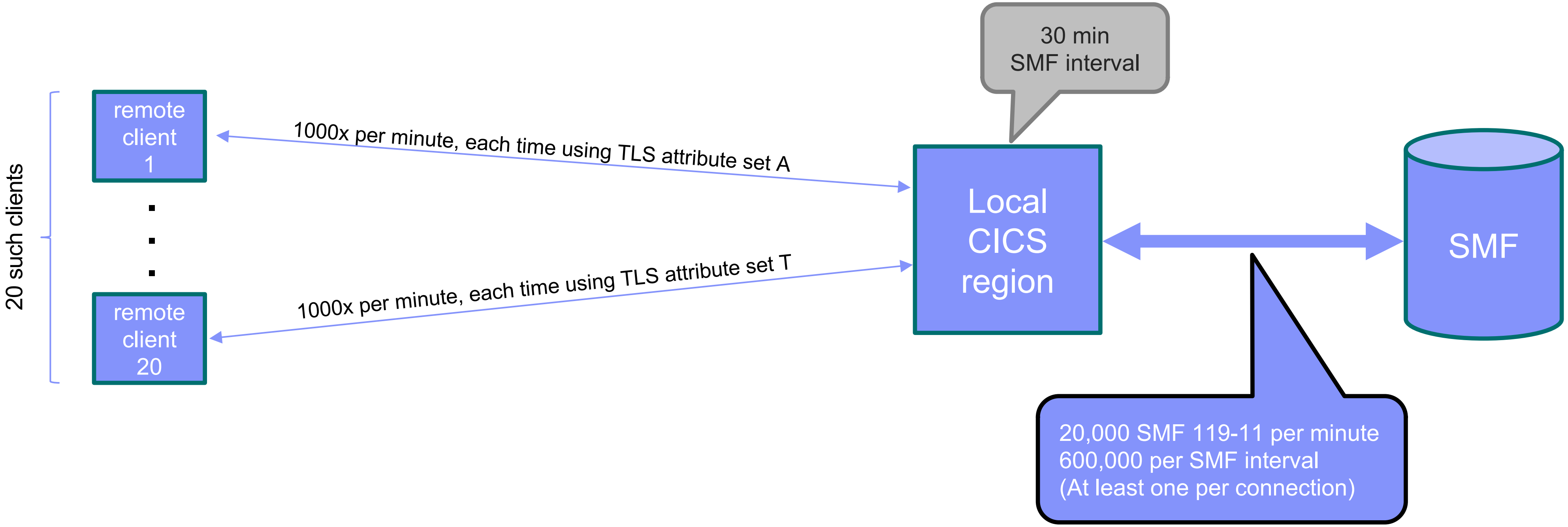
IPsec Protection Section (0 or 1)
IPsec protection details

X.509 Distinguished Name Section (0 or 1)
Subject and Issuer distinguished names from relevant certificates

Zero or more of these will be present

The need for zERT Aggregation

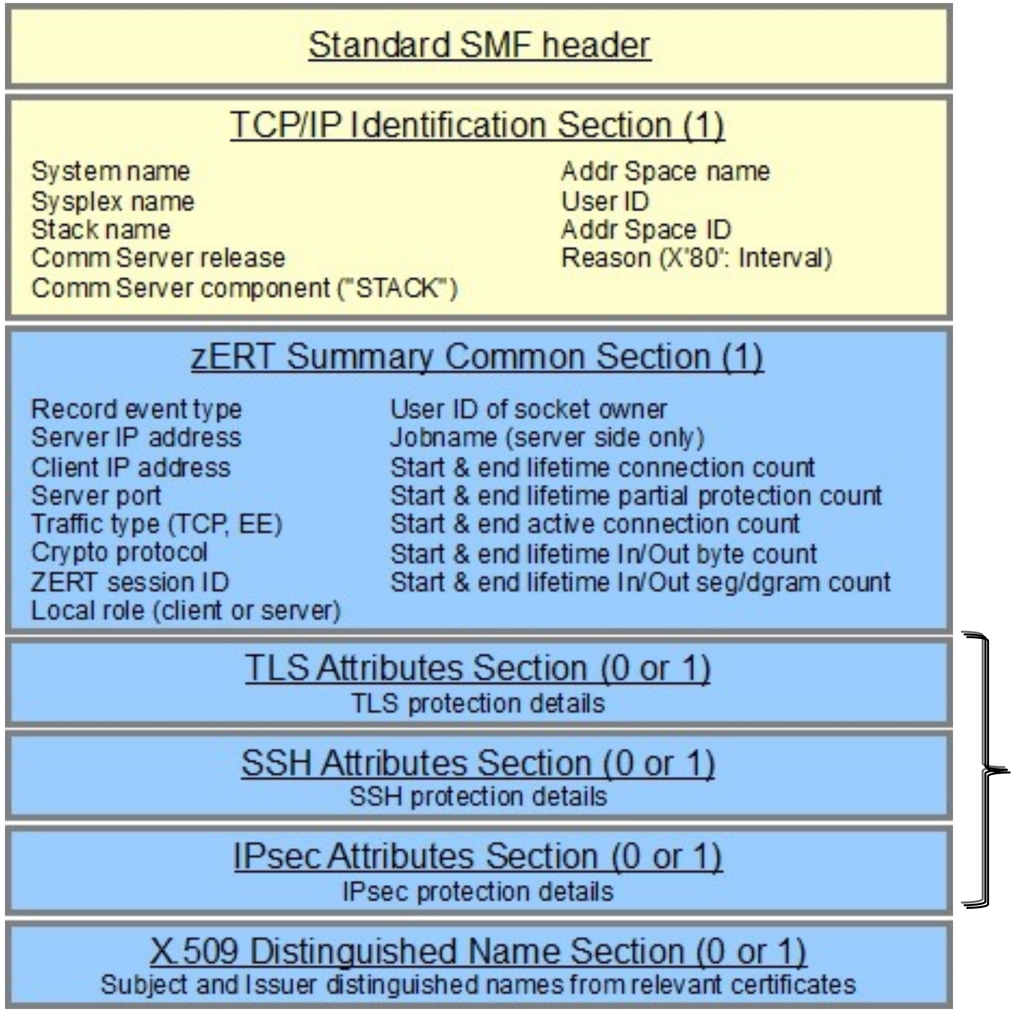
Workloads that consist of large numbers of frequent short-lived connections could generate huge volumes of zERT subtype 11 records


30 min
SMF interval
20 such clients
remote client 1
remote client 20
1000x per minute, each time using TLS attribute set A
1000x per minute, each time using TLS attribute set T
Local CICS region
SMF
20,000 SMF 119-11 per minute
600,000 per SMF interval
(At least one per connection)

Some measures are already taken in zERT Discovery to reduce the number of subtype 11 records (timers and "Short-lived Connection Termination" records), but in environments that manage thousands of connections per hour or minute, the number of subtype 11 records can still be very large

zERT Aggregation: SMF Summary record (type 119, subtype 12)

- **zERT Aggregation summarizes the repetitive use of security sessions over time**
  - From the server's perspective (based on server IP address, server port, & client IP address)
  - Regardless of whether z/OS is the client or the server
- **One Summary record is written at the end of each *recording* interval for each active security session**. Contains:
  - Connection attributes (server IP addr, server port, client IP addr, transport protocol)
  - *Significant* security attributes
  - Statistics (connection counts, byte counts, etc.)
- With aggregation, **the same example scenario from the previous page would result in 20 SMF 119 subtype 12 records per interval** – one per client TLS session
- Since SMF 119-12 record focus on the security session, each is associated with at most one security protocol. So cases of double protection (TLS + IPsec, for example) generate two SMF 119-12s since two different security sessions exist for a single connection.



Standard SMF header

TCP/IP Identification Section (1)

System name
Sysplex name
Stack name
Comm Server release
Comm Server component ("STACK")

Addr Space name
User ID
Addr Space ID
Reason (X'80': Interval)

zERT Summary Common Section (1)

Record event type
Server IP address
Client IP address
Server port
Traffic type (TCP, EE)
Crypto protocol
ZERT session ID
Local role (client or server)

User ID of socket owner
Jobname (server side only)
Start & end lifetime connection count
Start & end lifetime partial protection count
Start & end active connection count
Start & end lifetime In/Out byte count
Start & end lifetime In/Out seg/dgram count

TLS Attributes Section (0 or 1)
TLS protection details

SSH Attributes Section (0 or 1)
SSH protection details

IPsec Attributes Section (0 or 1)
IPsec protection details

X.509 Distinguished Name Section (0 or 1)
Subject and Issuer distinguished names from relevant certificates

Zero or one of these will be present

Configuring zERT in the TCPIP profile data set

zERT **in-memory collection enabled independently of destinations** to which records are written

- GLOBALCONFIG ZERT controls zERT in-memory monitoring
  - `GLOBALCONFIG ZERT [AGGRegation [agg-subparms]] | NOZERT` (Default is NOZERT)

- SMFCONFIG controls writing of zERT records to System Management Facility
  - `SMFCONFIG TYPE119 ZERTDetail | NOZERTDetail` (Default is NOZERTDetail)
  - `SMFCONFIG TYPE119 ZERTDetailByPolicy | NOZERTDetailByPolicy` (Default is NOZERTDetailByPolicy)
  - `SMFCONFIG TYPE119 ZERTSUMmary | NOZERTSUMmary` (Default is NOZERTSummary)

- NETMONITOR controls writing of zERT records to real-time network monitoring services
  - `NETMONITOR ZERTService | NOZERTService` (Default is NOZERTService)
  - `NETMONITOR ZERTServiceByPolicy | NOZERTServiceByPolicy` (Default is NOZERTServiceByPolicy)
  - `NETMONITOR ZERTSUMmary | NOZERTSUMmary` (Default is NOZERTSummary)

- All parameters can be dynamically enabled or disabled

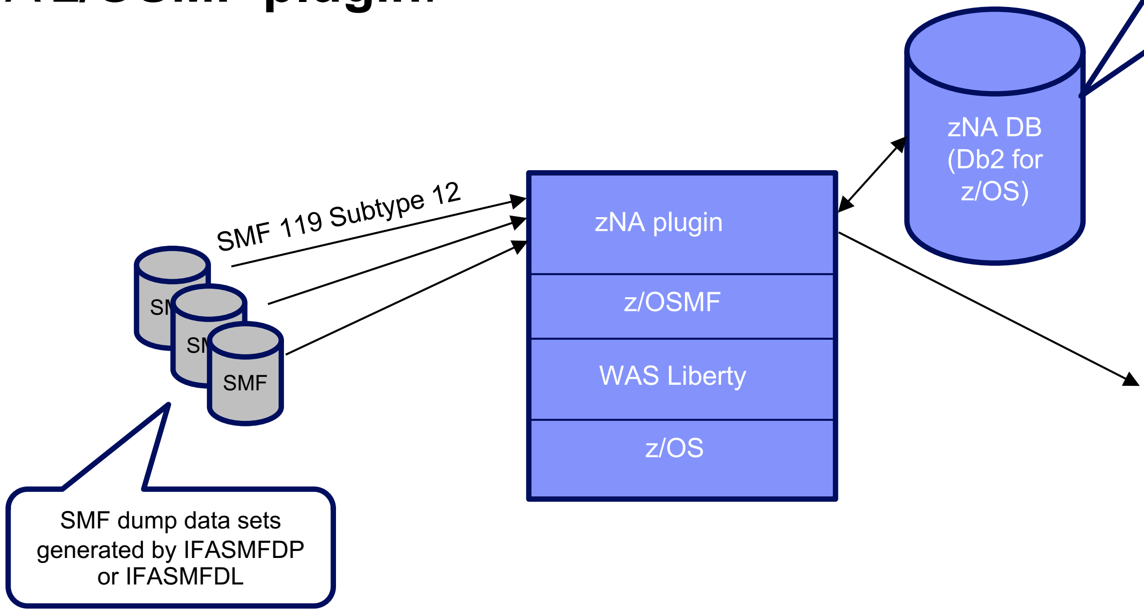
zERT Network Analyzer

- A **z/OSMF plugin**:

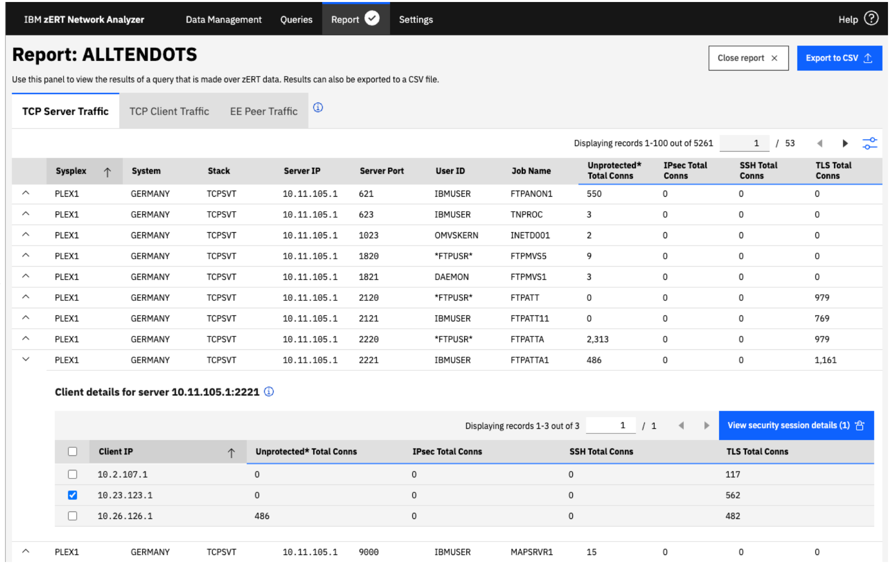
Contains zERT summary data for some fixed range of time

Users build queries over a variety of attributes:

- Specific scope attributes like Sysplex / system / stack, IP addresses / server port, z/OS role (client or server) and range of dates
- Specific security attributes like crypto protocol, protocol version, crypto algorithms and key lengths, etc.



SMF dump data sets generated by IFASMFDP or IFASMFDL



Report: ALLTENDOTS

Use this panel to view the results of a query that is made over zERT data. Results can also be exported to a CSV file.

Displaying records 1-100 out of 5261 1 / 53

	Sysplex	System	Stack	Server IP	Server Port	User ID	Job Name	Unprotected* Total Conns	IPsec Total Conns	SSH Total Conns	TLS Total Conns
^	PLEX1	GERMANY	TCPSVT	10.11.105.1	621	IBMUSER	FTPANON1	550	0	0	0
^	PLEX1	GERMANY	TCPSVT	10.11.105.1	623	IBMUSER	TNPROC	3	0	0	0
^	PLEX1	GERMANY	TCPSVT	10.11.105.1	1023	OMVSKERN	INETD001	2	0	0	0
^	PLEX1	GERMANY	TCPSVT	10.11.105.1	1820	*FTPUSR*	FTPMVS5	9	0	0	0
^	PLEX1	GERMANY	TCPSVT	10.11.105.1	1821	DAEMON	FTPMVS1	3	0	0	0
^	PLEX1	GERMANY	TCPSVT	10.11.105.1	2120	*FTPUSR*	FTPATT	0	0	0	979
^	PLEX1	GERMANY	TCPSVT	10.11.105.1	2121	IBMUSER	FTPATT11	0	0	0	769
^	PLEX1	GERMANY	TCPSVT	10.11.105.1	2220	*FTPUSR*	FTPATTA	2,313	0	0	979
v	PLEX1	GERMANY	TCPSVT	10.11.105.1	2221	IBMUSER	FTPATTA1	486	0	0	1,161

Client details for server 10.11.105.1:2221

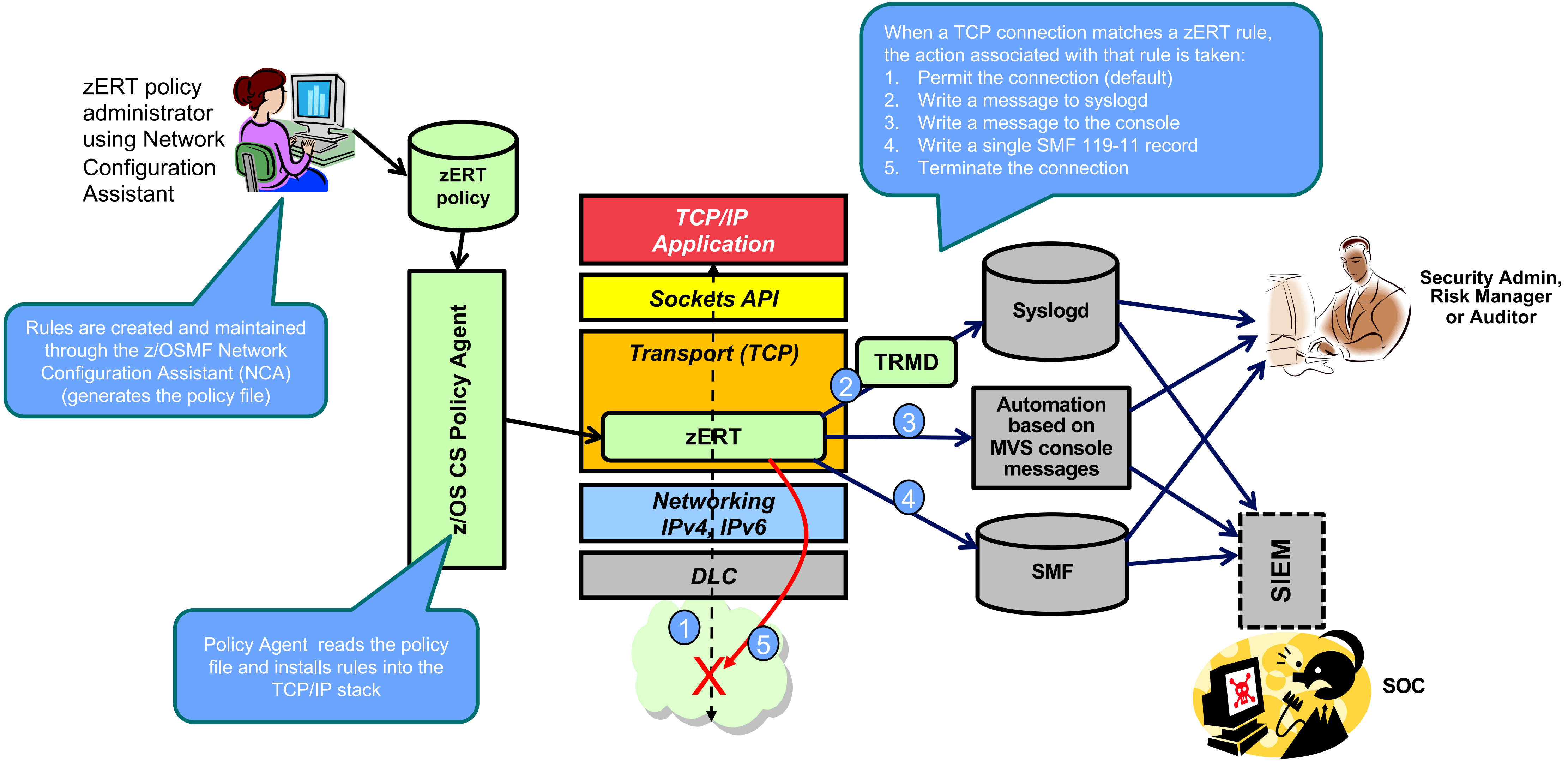
Displaying records 1-3 out of 3 1 / 1

□	Client IP	Unprotected* Total Conns	IPsec Total Conns	SSH Total Conns	TLS Total Conns
□	10.2.107.1	0	0	0	117
☑	10.23.123.1	0	0	0	562
□	10.26.126.1	486	0	0	482

	Sysplex	System	Stack	Server IP	Server Port	User ID	Job Name	Unprotected* Total Conns	IPsec Total Conns	SSH Total Conns	TLS Total Conns
^	PLEX1	GERMANY	TCPSVT	10.11.105.1	9000	IBMUSER	MAPSRVR1	15	0	0	0

- **Web UI** makes zERT data consumable for **z/OS network security administrators** (typically systems programmers)
- **Access to UI controlled through SAF** resource IZUDFLT.ZOSMF.ZERT_NETWORK_ANALYZER in the ZMFAPLA class
- Used primarily to investigate specific network encryption questions (but could also be used for periodic report generation)
- The latest zERT Network Analyzer PTF always has full install image

zERT policy-based enforcement


zERT policy administrator using Network Configuration Assistant
zERT policy
z/OS CS Policy Agent
TCP/IP Application
Sockets API
Transport (TCP)
zERT
Networking IPv4, IPv6
DLC
TRMD
Syslogd
Automation based on MVS console messages
SMF
SIEM
Security Admin, Risk Manager or Auditor
SOC
1
2
3
4
5

When a TCP connection matches a zERT rule, the action associated with that rule is taken:

1. Permit the connection (default)
2. Write a message to syslogd
3. Write a message to the console
4. Write a single SMF 119-11 record
5. Terminate the connection

Rules are created and maintained through the z/OSMF Network Configuration Assistant (NCA) (generates the policy file)

Policy Agent reads the policy file and installs rules into the TCP/IP stack

zERT Enforcement rules: General info

- Up to four separate “sets” of rules:
  - TLS/SSL
  - IPsec
  - SSH
  - No recognized protection (NRP)
- A single connection is evaluated against the zERT rules governing whichever security protocols are used for that connection (including “no recognized protection” rules)
  - One connection can match multiple rules (one per protocol)
  - If a connection does not match any rule, it is allowed (implicit “allow all” rule)
  - Specific events drive evaluation or re-evaluation of a connection against a given rule set
- Network Configuration Assistant guides you in the creation of these rule sets (V2R5 APAR PH35304)

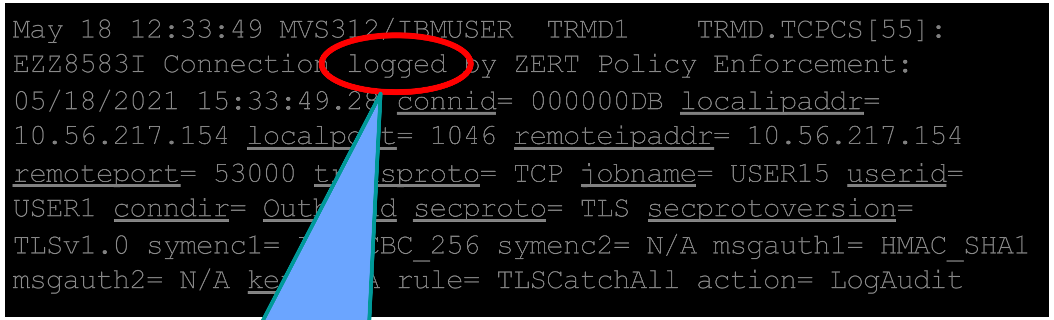
zERT Enforcement rules: Conditions

A zERT rule can be defined with the following conditions:

- Traffic attributes (specific rules only)
  - Local, remote IP addresses and ports
  - Jobname
  - z/OS user ID (that opened the socket)
  - Connection direction
  - TCP traffic only (EE support not currently planned)
- Protection attributes:
  - Protection protocol (TLS/SSL, IPsec, SSH, No Recognized Protection)
  - Protocol version (for TLS/SSL and SSH)
  - Symmetric encryption algorithms (including key lengths)
  - Message authentication/integrity algorithms (including key lengths)
  - Key exchange algorithms
  - In V2R5, zERT enforcement will NOT include digital signature algorithms or key lengths
- Time/Date when the rule is to be activated

zERT Enforcement rules: Actions

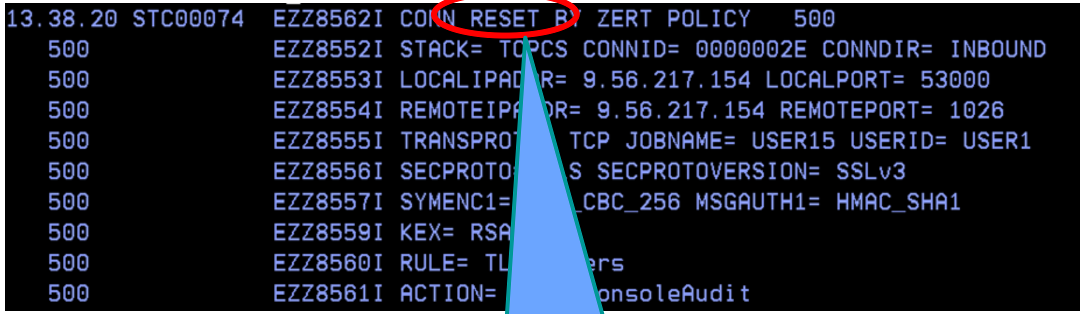
- Default action: Silently allow the TCP connection to proceed
- Reset (drop) TCP connection
- Reporting actions (can be specified in any combination, including the Reset action):
- Log to syslogd (subject to suppression to avoid flooding)



```
May 18 12:33:49 MVS312/IBMUSER  TRMD1    TRMD.TCPCS[55]:
EZZ8583I Connection logged by ZERT Policy Enforcement:
05/18/2021 15:33:49.28 connid= 000000DB localipaddr=
10.56.217.154 localpo  t= 1046 remoteipaddr= 10.56.217.154
remoteport= 53000 t   sproto= TCP jobname= USER15 userid=
USER1 conndir= Out    d secproto= TLS secprotoversion=
TLSv1.0 symenc1=    CBC_256 symenc2= N/A msgauth1= HMAC_SHA1
msgauth2= N/A ke      A rule= TLSCatchAll action= LogAudit
```

This rule specified log to syslogd action but not the reset action

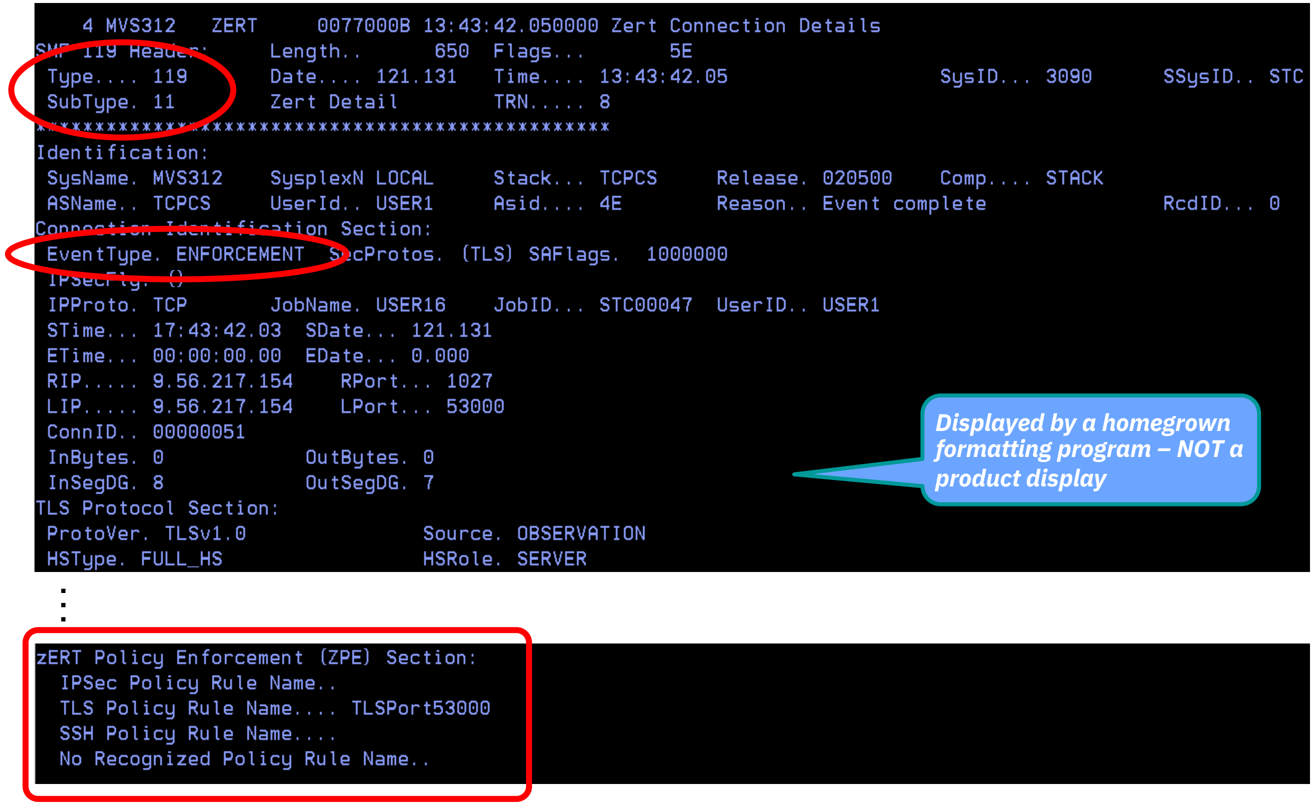
- Log to the console (subject to suppression to avoid flooding)



```
13.38.20 STC00074  EZZ8562I CONN RESET BY ZERT POLICY   500
   500             EZZ8552I STACK= TCPCS CONNID= 0000002E CONNDIR= INBOUND
   500             EZZ8553I LOCALIPAD R= 9.56.217.154 LOCALPORT= 53000
   500             EZZ8554I REMOTEIPA DR= 9.56.217.154 REMOTEPORT= 1026
   500             EZZ8555I TRANSPROT    TCP JOBNAME= USER15 USERID= USER1
   500             EZZ8556I SECPROTO    S SECPROTOVERSION= SSLv3
   500             EZZ8557I SYMENC1=     _CBC_256 MSGAUTH1= HMAC_SHA1
   500             EZZ8559I KEX= RSA
   500             EZZ8560I RULE= TL      ers
   500             EZZ8561I ACTION=       onsoleAudit
```

This rule specified log to console and reset actions

- Write an audit record (SMF type 119, subtype 11, event type 7 – enabled separately from all other SMF 119-11 records



```
    4 MVS312   ZERT     0077000B 13:43:42.050000 Zert Connection Details
SMF 119 Header:      Length..      650  Flags...       5E
 Type.... 119        Date.... 121.131   Time.... 13:43:42.05           SysID... 3090      SSysID.. STC
 SubType. 11         Zert Detail        TRN..... 8
*************************************************
Identification:
 SysName. MVS312     SysplexN LOCAL     Stack... TCPCS     Release. 020500    Comp.... STACK
 ASName.. TCPCS      UserId.. USER1     Asid.... 4E        Reason.. Event complete          RcdID... 0
Connection Identification Section:
 EventType. ENFORCEMENT  SecProtos. (TLS) SAFlags.  1000000
 IPSecFlg. ()
 IPProto. TCP        JobName. USER16    JobID... STC00047  UserID.. USER1
 STime... 17:43:42.03  SDate... 121.131
 ETime... 00:00:00.00  EDate... 0.000
 RIP..... 9.56.217.154    RPort... 1027
 LIP..... 9.56.217.154    LPort... 53000
 ConnID.. 00000051
 InBytes. 0            OutBytes. 0
 InSegDG. 8            OutSegDG. 7
TLS Protocol Section:
 ProtoVer. TLSv1.0               Source. OBSERVATION
 HSType. FULL_HS                 HSRole. SERVER
  :
zERT Policy Enforcement (ZPE) Section:
  IPSec Policy Rule Name..
  TLS Policy Rule Name.... TLSPort53000
  SSH Policy Rule Name....
  No Recognized Policy Rule Name..
```

Displayed by a homegrown formatting program – NOT a product display

Overview: zERT support in other products (as of October 2022)

IBM is aware of the following products that have shipped support for zERT data. Note that this should not be considered to be a comprehensive list as there may be others of which IBM is currently unaware:

- IBM zSecure Audit V2.3 (subtype 11 and subtype 12 records)
- IBM QRadar SIEM (supports what zSecure feeds it)
- Merrill Technologies MXG (feeds subtype 11 and subtype 12 records into SAS)
- Broadcom NetMaster Network Management for TCP/IP 12.2.03 (subtype 11 records through NMI)
- BMC Mainview for IP 3.6 (subtype 11 and subtype 12 records through NMI)
- Vanguard Advisor 2.3 (subtype 11 records)
- IntelliMagic Vision (subtype 12 records)
- IBM Z Common Data Provider 2.1.0 (subtype 11 and 12 records)
- IBM NetView Version 6.3 (supports subtype 11 records through NMI)
- IBM Omegamon for Networks on z/OS version 550, fixpack 4 (APAR OA57939 - subtype 11 records through NMI)
- Pacific Systems Group’s Spectrum SMF Writer (subtype 11 and 12 records)
- IBM Z Performance and Capacity Analytics V3.1.0 with APAR PH12196 (subtype 11 and 12 records)

For more information

URL	Content
http://ibm.biz/thingsaboutzert	IBM zERT “all-in-one” page
https://www.ibm.com/community/z/software/comm-server	IBM Communications Server blog
https://www.ibm.com/docs/en/zos/3.1.0?topic=zos-communications-server	IBM Communications Server library

Demo: zERT Network Analyzer

Thank you!