ExaCC Application Server Discovery and Firewall Planning for ExaCS Migration

Oracle ExaCC-side discovery approach for identifying application clients before cloud migration

Purpose

This document explains how to use the existing Oracle ExaCC database environment to identify application servers that currently connect to the database, collect source IP and service information, and build a firewall-rule matrix for a future ExaCS migration.

1. Can discovery be run from the ExaCC database?

Yes. Running discovery from ExaCC is a very useful first step because the database and listener already see the clients that successfully connect today. From ExaCC you can identify application hostnames, client IP addresses, Oracle usernames, services, session counts, and current TCP connections.

However, ExaCC-side discovery cannot reliably identify every physical firewall or the exact firewall rule number in the path. The connection is visible to ExaCC only after it has traversed the network. Firewall topology and policy must therefore be correlated with network/security tools such as Tufin, AlgoSec, FireMon, Palo Alto Panorama, Check Point SmartConsole, or firewall logs.

2. Identify connected application servers using GV$SESSION

Run the following query from the ExaCC database:

```
select
    machine,
    program,
    username,
    service_name,
    count(*) sessions
from gv$session
where username is not null
group by
    machine,
    program,
    username,
    service_name
order by machine;
```

This query provides a current snapshot of client machines connected to the database. It is especially useful for identifying application, batch, web, reporting, ETL, and integration hosts.

3. Collect Oracle connection metadata

```
select
    s.inst_id,
    s.username,
    s.machine,
    s.program,
    s.service_name,
    c.client_oci_library,
    c.network_service_banner
from gv$session s
left join gv$session_connect_info c
    on s.inst_id = c.inst_id
   and s.sid = c.sid
where s.username is not null
order by s.machine;
```

This gives additional connection metadata such as client library information and can help distinguish JDBC, OCI, ODP.NET, SQL*Plus, batch tools, and other client types.

4. Use listener logs to identify source IP addresses

Listener logs are one of the most valuable sources for migration discovery because they can retain historical connection attempts, including clients that are not connected at the exact moment you query GV$SESSION.

Typical listener log location:

```
$ORACLE_BASE/diag/tnslsnr/<hostname>/<listener_name>/trace/listener.log
```

Examples:

```
grep "HOST=" listener.log
```

```
grep -o 'HOST=[^)]*' listener.log | sort -u
```

Example source addresses discovered:

```
HOST=10.20.10.21
HOST=10.20.10.22
HOST=10.20.30.15
HOST=10.20.40.50
```

5. Check live TCP connections from the ExaCC OS

On the ExaCC database node, where access and support policy permit, use:

```
ss -tn | grep ':1521'
```

```
netstat -tn | grep ':1521'
```

Example:

```
ESTAB 0 0 10.50.1.20:1521 10.20.10.21:53144
ESTAB 0 0 10.50.1.20:1521 10.20.10.22:49831
```

In this example, 10.50.1.20 is the ExaCC listener/database-side IP and 10.20.10.21 / 10.20.10.22 are application-side client IPs.

6. Build an application-client inventory

Application Host	Client IP	DB Service	Port	Sessions
APP01	10.20.10.21	PROD_APP	1521	43
APP02	10.20.10.22	PROD_APP	1521	41
BATCH01	10.20.30.15	PROD_BATCH	1521	7

This inventory becomes the starting point for the future ExaCS firewall design.

7. Convert the current inventory into the future ExaCS firewall matrix

For each discovered application source, replace the current ExaCC destination with the future ExaCS SCAN/VIP/service endpoint.

```
APP01  10.20.10.21  ->  ExaCS SCAN/VIP  -> TCP/1521
APP02  10.20.10.22  ->  ExaCS SCAN/VIP  -> TCP/1521
WEB01  10.20.20.11  ->  ExaCS SCAN/VIP  -> TCP/1521
BATCH01 10.20.30.15 ->  ExaCS SCAN/VIP  -> TCP/1521
```

8. Recommended firewall rule matrix

Source	Source IP	Destination	Destination IP	Port	Protocol	Direction	Purpose
APP01	10.20.10.21	ExaCS PROD	10.150.30.100	1521	TCP	APP -> DB	Oracle
APP02	10.20.10.22	ExaCS PROD	10.150.30.100	1521	TCP	APP -> DB	Oracle
WEB01	10.20.20.11	ExaCS PROD	10.150.30.100	1521	TCP	APP -> DB	Oracle

9. Important limitation: current sessions are not enough

Do not rely only on a single GV$SESSION snapshot. Some systems connect only during scheduled jobs, month-end processing, overnight batches, reporting windows, backup workflows, or infrequent integrations.

For migration planning, use several days or preferably 30-60 days of listener/firewall history, together with current GV$SESSION and OS-level TCP connection information.

10. Recommended discovery sources

Source	What it tells you	Why it matters

GV$SESSION	Currently connected application hosts, users, services	Fast snapshot of live clients
GV$SESSION_CONNECT_INFO	Client library and connection metadata	Helps classify application technology
Listener logs	Historical client hosts/IPs and connection attempts	Finds intermittent/batch clients
ss / netstat	Current TCP source/destination pairs	Confirms active network endpoints
Firewall/flow logs	Allowed and denied historical flows	Best evidence for security rule planning
Tufin / AlgoSec / FireMon	Firewall path and policy analysis	Identifies enforcement points and matching rules

11. Recommended migration workflow

1. Query GV$SESSION to identify currently connected clients.

2. Review listener logs over 30-60 days to identify intermittent clients.

3. Use ss/netstat for live TCP source IP confirmation.

4. Resolve application hostnames to source IP/subnet.

5. Build the current ExaCC source-to-destination matrix.

6. Replace ExaCC destination details with future ExaCS SCAN/VIP/service details.

7. Give the matrix to the network/security team.

8. Use Tufin, AlgoSec, FireMon, Panorama, SmartConsole, or firewall logs to trace the required policies.

9. Implement firewall and OCI NSG/Security List rules well before migration.

10. Validate from the real application servers using nc, TCP traceroute, tnsping, and SQL*Plus.

11. Repeat GV$SESSION validation after cutover to confirm all expected application clients are connected to ExaCS.

12. Key conclusion

Running discovery from ExaCC is highly recommended because it provides a central view of the clients that actually use the current database. It is an excellent way to build the application-server inventory and firewall requirements for ExaCS. It should be combined with historical listener/firewall logs and network policy-analysis tools because the ExaCC server itself cannot reliably identify every firewall or rule that traffic traverses.

Prepared as a practical pre-migration firewall discovery guide for Oracle ExaCC -> ExaCS.