



Schema: User Activity Tables

This chapter contains information on the schema and supported joins for user activity and identity events. The Firepower System can detect user activity on your network by tracking various types of user logins, including LDAP, POP3, IMAP, SMTP, AIM, and SIP.

For more information, see the sections listed in the following table.

Table 8-1 Schema for User Identity Tables

| See... | For the table that stores information on... | Version |
|--|--|---------|
| discovered_users, page 8-1 | Information about the users detected by the system. | 5.0+ |
| user_discovery_event, page 8-2 | User discovery events, which communicate the details of user activity on your network. | 5.0+ |
| user_ioc_state, page 8-4 | Stores compromise state for users. | 6.2+ |

discovered_users

The `discovered_users` table contains detailed information about each user detected by the system.

The `discovered_users` table supersedes the deprecated `rua_user` table starting with Version 5.0 of the Firepower System.

For more information, see the following sections:

- [discovered_users Fields, page 8-1](#)
- [discovered_users Joins, page 8-2](#)
- [discovered_users Sample Query, page 8-2](#)

discovered_users Fields

The following table describes the fields you can access in the `discovered_users` table.

Table 8-2 `discovered_users` Fields

| Field | Description |
|-------|---------------------------------|
| dept | The department of the user. |
| email | The email address for the user. |

Table 8-2 *discovered_users Fields (continued)*

| Field | Description |
|------------------|---|
| first_name | The first name for the user. |
| ip_address | This field has been deprecated and returns null for all queries. |
| ipaddr | A binary representation of the IPv4 or IPv6 address for the host where the user login was detected. |
| last_name | The last name for the user. |
| last_seen_sec | The UNIX timestamp of the date and time the system last reported a login for the user. |
| last_updated_sec | The UNIX timestamp of the date and time the user's information was last updated. |
| name | The name for the user. |
| phone | The phone number for the user. |
| rna_service | Field deprecated in Version 5.0. Returns null for all queries. |
| user_id | The internal identification number of the user who last logged onto the host. |

discovered_users Joins

The following table describes the joins you can perform on the `rna_user` table.

Table 8-3 *discovered_users Joins*

| You can left join on this field... | With other tables that have join type of... |
|------------------------------------|--|
| user_id | <code>user_discovery_event.user_id</code> <code>user_ipaddr_history.user_id</code> <code>user_ioc_state.user_id</code> |

discovered_users Sample Query

The following query returns up to 25 discovered user records that were generated since a specified date and time.

```
SELECT user_id, ip_address, email, name, last_seen_sec, last_updated_sec
FROM discovered_users
WHERE last_seen_sec >= UNIX_TIMESTAMP("2011-10-01 00:00:00")
LIMIT 0, 25;
```

user_discovery_event

The `user_discovery_event` table contains a record for each user discovery event.

Note that starting in Version 5.0, the Firepower System records the detection of user activity at the managed device level, no longer by detection engine. The `detection_engine_name` and `detection_engine_uuid` fields in this table have been replaced by the `sensor_name` and `sensor_uuid` fields respectively. Queries on these fields will return information about the managed device that generated the user discovery event.

For more information, see the following sections:

- [user_discovery_event Fields, page 8-3](#)
- [user_discovery_event Joins, page 8-4](#)
- [user_discovery_event Sample Query, page 8-4](#)

user_discovery_event Fields

The following table describes the fields you can access in the `user_discovery_event` table.

Table 8-4 *user_discovery_event Fields*

| Field | Description |
|--|--|
| <code>application_protocol_id</code> | An internal identifier for the detected application protocol. |
| <code>application_protocol_name</code> | One of: <ul style="list-style-type: none"> • the name of the application used in the connection: LDAP, POP3, and so on • <code>pending</code> if the system cannot identify the application for one of several reasons • blank if there is no application information in the connection |
| <code>description</code> | The user name when the discovery event type is either Delete User Identity, or User Identity Dropped. Otherwise, blank. |
| <code>domain_name</code> | Name of the domain for the on which the user was detected. |
| <code>domain_uuid</code> | UUID of the domain in which the user was detected. This is presented in binary. |
| <code>endpoint_profile</code> | Name of the type of device used by the connection endpoint. |
| <code>event_id</code> | An internal identification number for the discovery event. |
| <code>event_time_sec</code> | The UNIX timestamp of the date and time of the discovery event. |
| <code>event_type</code> | The type of discovery event. For example, <code>New User Identity</code> or <code>User Login</code> . |
| <code>ip_address</code> | Field deprecated in Version 5.2. Returns <code>null</code> for all queries. |
| <code>ipaddr</code> | A binary representation of the IP address of the host where the user activity was detected. |
| <code>location_ip</code> | IP address of the interface communicating with ISE. Can be IPv4 or IPv6. |
| <code>reported_by</code> | The IPv4 address, IPv6 address, or NetBIOS name of the Active Directory server reporting a user login. |
| <code>security_group</code> | ID number of the network traffic group. |
| <code>sensor_address</code> | The IP address of the managed device that detected the user discovery event. Format is <code>ipv4_address</code> , <code>ipv6_address</code> . |
| <code>sensor_name</code> | The text name of the managed device that detected the user discovery event. |
| <code>sensor_uuid</code> | A unique identifier for the managed device, or 0 if <code>sensor_name</code> is <code>null</code> . |
| <code>user_dept</code> | The department of the user who last logged onto the host. |
| <code>user_email</code> | The email address of the user who last logged onto the host. |
| <code>user_first_name</code> | The first name of the user. |
| <code>user_id</code> | The internal identification number of the user who last logged onto the host. |
| <code>user_last_name</code> | The last name of the user. |

Table 8-4 *user_discovery_event Fields (continued)*

| Field | Description |
|-----------------------|--|
| user_last_seen_sec | The UNIX timestamp of the date and time the system last reported a login for the user. |
| user_last_updated_sec | The UNIX timestamp of the date and time the user's information was last updated. |
| user_name | The user name for the user who last logged onto the host. |
| user_phone | The phone number for the user who last logged onto the host. |

user_discovery_event Joins

The following table describes the joins you can perform on the `user_discovery_event` table.

Table 8-5 *user_discovery_event Joins*

| You can join this table on... | And... |
|-------------------------------|--|
| ipaddr | <code>rna_host_ip_map.ipaddr</code> <code>user_ipaddr_history.ipaddr</code> |
| user_id | <code>discovered_users.user_id</code> <code>user_ipaddr_history.user_id</code> <code>user_ioc_state.user_id</code> |

user_discovery_event Sample Query

The following query returns up to 25 user event records generated by a selected managed device since a particular date and time.

```
SELECT event_time_sec, ipaddr, sensor_name, event_type, user_name, user_last_seen_sec,
user_last_updated_sec
FROM user_discovery_event
WHERE sensor_name = sensor_name
AND user_last_seen_sec >= UNIX_TIMESTAMP("2011-10-01 00:00:00") ORDER BY event_type ASC
LIMIT 0, 25;
```

user_ioc_state

The `user_ioc_state` table stores the IOC state for users in your monitored network.

For more information, see the following sections:

- [user_ioc_state Fields, page 8-5](#)
- [user_ioc_state Joins, page 8-7](#)
- [user_ioc_state Sample Query, page 8-7](#)

user_ioc_state Fields

The following table describes the fields you can access in the `user_ioc_state` table.

Table 8-6 *user_ioc_state Fields*

| Field | Description |
|--|--|
| <code>first_seen</code> | Unix timestamp when the compromise was first detected. |
| <code>first_seen_sensor_address</code> | The IP address of the managed device that first detected the compromise. Format is <i>ipv4_address, ipv6_address</i> . |
| <code>first_seen_sensor_name</code> | The managed device that first detected the compromise. |
| <code>user_id</code> | ID number of the user. |
| <code>ioc_category</code> | The category for the compromise. Possible values include: <ul style="list-style-type: none"> • CnC Connected • Exploit Kit • High Impact Attack • Low Impact Attack • Malware Detected • Malware Executed • Dropper Infection • Java Compromise • Word Compromise • Adobe Reader Compromise • Excel Compromise • PowerPoint Compromise • QuickTime Compromise |
| <code>ioc_description</code> | Description of the compromise. |

Table 8-6 user_ioc_state Fields (continued)

| Field | Description |
|----------------|--|
| ioc_event_type | <p>The event type for the compromise. Possible values include:</p> <ul style="list-style-type: none"> • Adobe Reader launched shell • Dropper Infection Detected by AMP for Endpoints • Excel Compromise Detected by AMP for Endpoints • Excel launched shell • Impact 1 Intrusion Event – attempted-admin • Impact 1 Intrusion Event – attempted-user • Impact 1 Intrusion Event – successful-admin • Impact 1 Intrusion Event – successful-user • Impact 1 Intrusion Event – web-application-attack • Impact 2 Intrusion Event – attempted-admin • Impact 2 Intrusion Event – attempted-user • Impact 2 Intrusion Event – successful-admin • Impact 2 Intrusion Event – successful-user • Impact 2 Intrusion Event – web-application-attack • Intrusion Event – exploit-kit • Intrusion Event – malware-backdoor • Intrusion Event – malware-CnC • Java Compromise Detected by AMP for Endpoints • Java launched shell • PDF Compromise Detected by AMP for Endpoints • PowerPoint Compromise Detected by AMP for Endpoints • PowerPoint launched shell • QuickTime Compromise Detected by AMP for Endpoints • QuickTime launched shell • Security Intelligence Event – CnC • Suspected Botnet Detected by AMP for Endpoints • Threat Detected by AMP for Endpoints – Subtype is 'executed' • Threat Detected by AMP for Endpoints – Subtype is not 'executed' • Threat Detected in File Transfer – Action is not 'block' • Word Compromise Detected by AMP for Endpoints • Word launched shell |
| ioc_id | Unique ID number for the compromise. |
| is_disabled | Whether this compromise has been disabled. |
| last_seen | Unix timestamp when this compromise was last detected. |

Table 8-6 *user_ioc_state Fields (continued)*

| Field | Description |
|--------------------------|---|
| last_seen_sensor_address | The IP address of the managed device that last detected the compromise. Format is <i>ipv4_address, ipv6_address</i> . |
| last_seen_sensor_name | The managed device that last detected the compromise. |

user_ioc_state Joins

The following table describes the joins you can perform on the `user_ioc_state` table.

Table 8-7 *user_ioc_state Joins*

| You can join this table on... | And... |
|-------------------------------|--|
| user_id | <code>discovered_users.user_id</code> <code>user_ipaddr_history.user_id</code> <code>user_discovery_event.user_id</code> |

user_ioc_state Sample Query

The following query returns up to 25 hosts with their ioc within a specified timespan.

```
SELECT user_id, ioc_id
FROM user_ioc_state
WHERE first_seen
BETWEEN UNIX_TIMESTAMP("2011-10-01 00:00:00")
AND UNIX_TIMESTAMP("2011-10-07 23:59:59")
ORDER BY ioc_id DESC
LIMIT 0, 25;
```

