



Use SQL to corrupt their databases...

# LET'S HACK ROCK

# COURTNEY COOKSEY

Software Engineer

The Crossing EPC

Columbia, MO

Enid, OK



MADE WITH

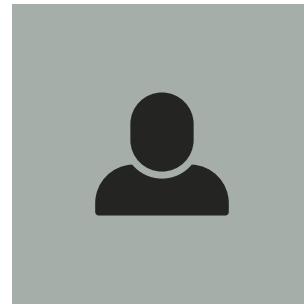
**beautiful.ai**

# AGENDA

- 1 Rock Security 101
- 2 Person Merge Attack
- 3 Account Takeover Attack
- 4 Pitfalls with Impersonation Tokens
- 5 SQL Injection Attack
- 6 How You can Protect Your System

# ROCK SECURITY 101

# THERE ARE TWO MAIN ENTITIES INVOLVED IN AUTHENTICATION



## Person Entity

Rules for what someone can access is tied to their person record



## User Login Entity

User logins are used to verify that a specific person record belongs to you

# ACCOUNT PROTECTION LEVELS

1

Low

No risk items

2

Medium

Has login account

3

High

Active Scheduled Financial  
Transaction

Saved Payment Account

In a Security Role Marked w/ High  
Elevated Security

4

Extreme

In a Security Role Marked w/ Extreme  
Elevated Security

# SECURITY SETTINGS FOR ACCOUNT PROTECTION LEVELS

---

Disable Duplicate Checking for the Following Protection Profiles [?](#)

Low

Medium

High

Extreme

Allow Merges of Account Protection Profile - High [?](#)

RSR - Data Integrity Worker

Disable Usage of Personal Tokens for the Following Protection Profiles [?](#)

Low

Medium

High

Extreme

# PERSON MERGE ATTACK

ALL SYSTEMS VULNERABLE

# TIMELINE OF ATTACK



# SELECTING A TARGET



- Ideal target is a millennial or active on social media
- Someone with a public email or who's email we can obtain
- Someone we can easily obtain more data on via Google

# Kristin Jeffries

Creative Arts Co-Director

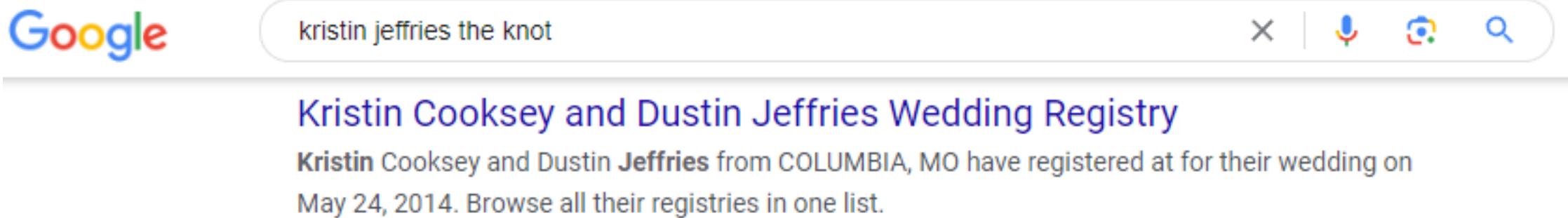
[kristin@thecrossingchurch.com](mailto:kristin@thecrossingchurch.com)

Kristin was born and raised in Columbia. She began attending The Crossing as a high school student and became involved with the preschool and elementary large group teams while studying theatre at Stephens College. Through many years of serving in Crossing Kids, including three years as an intern, it became clear that Kristin had a heart for children's ministry. After marrying her husband, Dustin, in May of 2014, she left the world of theatre and dance to join the Crossing Kids team full-time in January of 2015.

As the Early Childhood Programming Director, Kristin oversees the preschool and little ones' teaching and worship teams, assists with family events, and creates early childhood curriculum. She loves leading worship and being involved in family events.

# LOOK FOR A MAIDEN NAME

We have her current name, her husband's name, and the month and year they were married. With a few checks of the most popular wedding websites we get a hit



A screenshot of a Google search results page. The search bar at the top contains the query "kristin jeffries the knot". Below the search bar, the first result is a link titled "Kristin Cooksey and Dustin Jeffries Wedding Registry". The snippet below the title reads: "Kristin Cooksey and Dustin Jeffries from COLUMBIA, MO have registered at for their wedding on May 24, 2014. Browse all their registries in one list." The Google logo is visible on the left side of the search bar.

Google

kristin jeffries the knot

[Kristin Cooksey and Dustin Jeffries Wedding Registry](#)

Kristin Cooksey and Dustin Jeffries from COLUMBIA, MO have registered at for their wedding on May 24, 2014. Browse all their registries in one list.

# LOOK FOR DATE OF BIRTH INFORMATION

Armed with her maiden name and hometown we can look for other information about her. We get a hit for a high school honor roll for both 2008 and 2009 which narrows down her birth year to being within 1990-1992

Google search results for "kristin cooksey columbia mo". The search bar shows the query. Below it, a news snippet from the Columbia Daily Tribune dated April 2, 2009, lists Kristin Cooksey as one of the recipients of the Rock Bridge High School Honor Roll for the First Semester.

kristin cooksey columbia mo

Columbia Daily Tribune  
<https://www.columbiatribune.com › lifestyle › 2009/04/02> ::

**HONOR ROLL: Rock Bridge High School, First Semester**

Apr 2, 2009 – Columbia Daily Tribune ... Peter Colman, Kristin Cooksey, Hannah Cossey, Tanner Craigmire, Tessa Crawford, ... 2023 www.columbiatribune.com.

# THE INTERNET IS A WEALTH OF PERSONAL INFORMATION



Name	Phone	Address	Email	Company
Kristin Cooksey				

**100% CONFIDENTIAL**

Kristin Cooksey  city, state

**Kristin Cooksey** • Age 32 / Jan 1991

**View Profile**

© Columbia, MO

**ALSO KNOWN AS**

Kristin M Cooksey • Kristin Jeffries

**RELATED TO**

Darrell Cooksey, 64 • Diana Cooksey, 64 • Courtney Cooksey • Dustin Jeffries, 35 • James Jeffries, 61 • Gemma Riccardi, 65 • Brian Cooksey, 37

**HAS LIVED IN**

Columbia, MO

# START TO CREATE A DUPLICATE ACCOUNT

We know her name, email, and two parts of her birth date. Let's see what happens when we create a new account in Rock with this information.

## Account Registration

There are already one or more people in our system that have the same email address and last name as you do. Are any of these people you?

You?	Name	Gender	Birth Day
<input type="radio"/>	Kristin Jeffries	Female	January 11

None of these are me

Previous

Next

# REGISTER OUR FAKE ACCOUNT

## Account Registration

New Account

Username •

kmjeffries

The selected username is available.

Password •

.....

Confirm Password •

.....

Your Information

First Name •

Kristin

Last Name •

Jeffries

Email •



kristin.jeffries11@gmail.com

Gender

Female



Birthday •

Jan



/

1



/ 1991

# UPDATE EMAIL OF FAKE ACCOUNT TO STAFF EMAIL

Photo



## My Account

### Personal Information

Title

First Name •

Last Name •

Nick Name

Suffix

Birthday

 /  / 

Gender •

Male  Female

### Contact Information

Email Address •

 kristin@thecrossingchurch.com

Account Info

Change Password

Giving

You currently have no active profiles.

Manage

View History

Give Now

Groups

MADE WITH

beautiful.ai

**NOW WE WAIT...**

# THE MERGE ALERT IS NOT NECESSARILY HELPFUL

We are merging a staff member's record, it is expected that they have a login associated with their record

Merge Records

Add Another Person

 ▾

**CRITICAL SECURITY ALERT:** One or more of the records has a login. This could be an attempt to hijack the account. Additionally, this person will be prompted to reconfirm before they can login.

\* ADDITIONALLY, ONE OR MORE OF THESE RECORDS IS A MEMBER OF A SECURITY ROLE WITH ELEVATED PRIVILEGES.

# VULNERABILITIES WHEN DUPLICATE CHECKING IS NOT DISABLED

We highly recommend that you prevent duplicate detection for individuals with an Account Protection Profile of High and Extreme

Disable Duplicate Checking for the Following Protection Profiles [i](#)

 Low Medium High Extreme

Allow Merges of Account Protection Profile - High [i](#)

RSR - Data Integrity Worker

Allow Merges of Account Prot

RSR - Rock Administratio

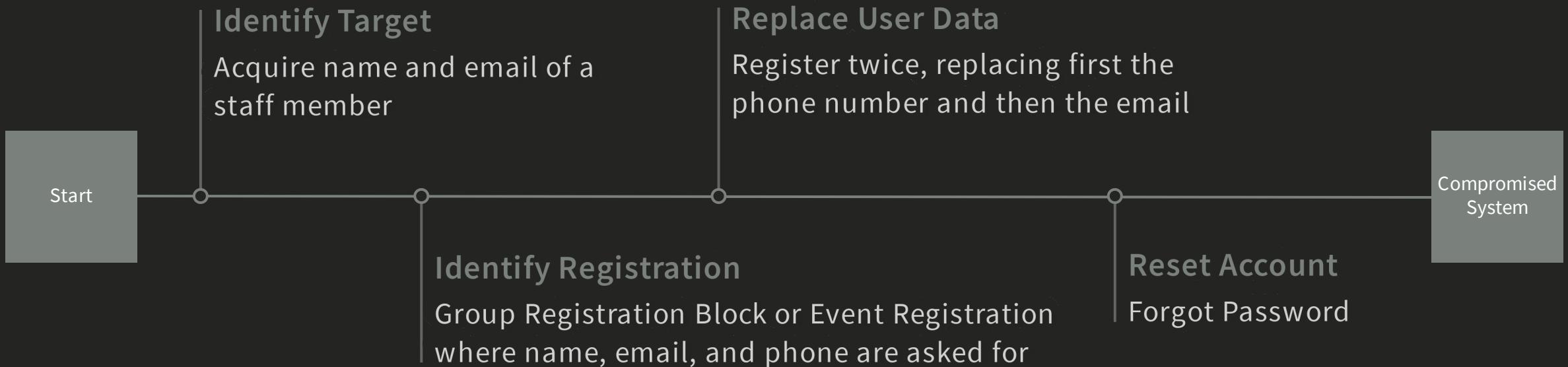
Disable Usage of Personal Tokens for the Following Protection Profiles [i](#)

 Low Medium High Extreme

# ACCOUNT TAKEOVER ATTACK

SYSTEMS WITHOUT DUPLICATE CHECKING DISABLED VULNERABLE

# TIMELINE OF ATTACK



# FIND AN EVENT OR GROUP REGISTRATION THAT ASKS FOR NAME, PHONE, AND EMAIL

## Person

First Name \*

Kristin

Last Name \*

Jeffries

Mobile \*

1 ▾ 573550102

Email \*



kristin@thecrossingchurch.com

# USING ANY REGISTRAR OPTION BESIDES USE FIRST REGISTRANT LETS US CHANGE ALL INFO

Registrar Options ⓘ

- Pre-fill First Registrant
- Prompt For Registrar
- Pre-fill First Registrant
- Use First Registrant
- Use Logged In Person

## Review Registration

This Registration Was Completed By

First Name

Last Name

Send Confirmation Emails To

**Kristin Jeffries** (Previous Names: Cooksey)

Member Staff Protection Profile: Extreme

32 yrs old (1/11/1991)  
Female  
Married 8 yrs (5/24)

(573) 555-0102 Mobile    
[kristin.jeffries11@gmail.com](mailto:kristin.jeffries11@gmail.com)

[Client Email](#)

# WE CAN DO THE SAME THING WITH GROUP REGISTRATION

- Register with correct name and email, new phone number
- Refresh Page
- Register with correct name and new phone number and email
- Person record in Rock is updated with new contact info

## Group Registration

Group Registration

Please complete the form below to register for Active Members & Attendees within Jeff City and surrounding area.

First Name \*

Kristin

Last Name \*

Jeffries

Home Phone

1 ▾

Mobile Phone

1 ▾ (573) 555-0102

Enable SMS

Email \*

 kristin@thecrossingchurch.com

Address

United States

Address

City

MO

Zip

Register

# ROCK WILL SEND THE ATTACKER AN EMAIL TO RESET THE USER LOGIN

From here we have however much time until our target tries to log in before someone notices something is wrong.



ROCK SANDBOX

Below are your current usernames at The Crossing

Username: testuseraccount

Reset Password

# PITFALLS WITH IMPERSONATION TOKENS

# PARAMETERS FOR THE PERSON TOKEN CREATE LAVA FILTER

- Minutes  
How long should the token be valid?
- Max Usage  
How many times can you use it?
- Page Id  
What page should it work for?

# THESE PARAMETERS ARE USED WHEN ESTABLISHING NEW SESSIONS

```
{{ Person | PersonTokenCreate:30,null,null }}
```

<https://rock.mychurch.com?rckipid=ACDC123>

30 minutes is the timeframe I can use that link to create a new session

A previously established session will not end after 30 minutes

# LET'S ASSUME WE HAVE TOKENS DISABLED FOR EXTREME PROFILES

RSR - Data Integrity Worker

---

Disable Usage of Personal Tokens for the Following Protection Profiles ?

Low       Medium       High       Extreme

# ISSUES WITH DISABLING IMPERSONATION

## Group Attendance Reminders

Please remember to enter attendance **for** your group meeting.

```
{{ 'Global' | Attribute:'PublicApplicationRoot' }}page/368?  
{{ Person.ImpersonationParameter }}&GroupId={{ Group.Id  
}}&Occurrence={{ Occurrence | Date:'yyyy-MM-  
ddTHH\%3amm\%3ass' }}
```

<https://rock.mychurch.com/page/368?rckipid=TokenProhibited>

# FIX BAD USER EXPERIENCE BY CONDITIONALLY ADDING TOKENS

```
{% if Person.AccountProtectionProfile != 'Extreme' %}{%
endif %}
```

# FIX BAD USER EXPERIENCE BY CONDITIONALLY ADDING TOKENS

```
{% if Person.AccountProtectionProfile != 'Extreme' %}{%
endif %}
```

Please remember to enter attendance for your group meeting.  
{{ 'Global' | Attribute:'PublicApplicationRoot' }}page/368?  
{% if Person.AccountProtectionProfile != 'Extreme' %}{{  
Person.ImpersonationParameter }}{% endif %}&GroupId={{  
Group.Id }}&Occurrence={{ Occurrence | Date:'yyyy-MM-  
ddTHH\%3amm\%3ass' }}

# SQL INJECTION ATTACK

ALL SYSTEMS VULNERABLE

# A BASIC DYNAMIC DATA BLOCK

```
{% assign searchOne = 'Global' | PageParameter:'Name' %}  
SELECT TOP 10 FirstName, LastName FROM Person WHERE FirstName LIKE '%{{searchOne}}%'
```

BlockTitle

Name Person

Test

Filter Reset Filters

Filter Options ▾

First Name	Last Name
Test	Cleverly
Test	Cleverly
Test1	Cleverly
Test2	Cleverly
Test3	Cleverly

50 500 5,000 5 Items

The screenshot shows a user interface for a dynamic data block. At the top, there's a search bar labeled 'Name' with 'Test' typed in, and a dropdown menu labeled 'Person' showing a person icon. Below this is a 'Filter' button and a 'Reset Filters' button. A 'Filter Options' dropdown is open. The main area displays a table with two columns: 'First Name' and 'Last Name'. The table contains six rows of data: 'Test' and 'Cleverly', 'Test' and 'Cleverly', 'Test1' and 'Cleverly', 'Test2' and 'Cleverly', and 'Test3' and 'Cleverly'. At the bottom left, there are pagination controls: '50', '500', '5,000', and '5 Items'. The entire interface is contained within a light gray box.

# A SIMPLE TABLE IN OUR DATABASE

SQL Command

SQL Text 

1 `SELECT * FROM TestTable`

Selection Query? 

No Yes

Run

column\_1

1

2

column\_2

1

2

50 500 5,000

2 Items

MADE WITH

beautiful.ai

# PERFORMING A SQL INJECTION ATTACK

BlockTitle

Name

```
'; DROP TABLE TestTable; SELECT '
```

[Filter](#) [Reset Filters](#)

Name='; DROP TABLE TestTable; SELECT '

# OUTPUT OF EXECUTING THIS ATTACK

First Name	Last Name
Taylor	
DaeJera	Aaron
Joy	Aaron
Ann	Abare
Tom	Abare
Ashley	Abast
Jax	Abbadessa
Bella	Abbell
Mimi	Abbet
Amanda	Abbey

50    500    5,000    10 Items



Column 1

%

MADE WITH

beautiful.ai

# QUERY AFTER TABLE DROP IS PERFORMED

SQL Text 

1 SELECT \* FROM TestTable

Selection Query? 

No Yes

Run

Error SQL Error! Invalid object name 'TestTable'.

# WHAT WAS ACTUALLY RUN

```
SELECT TOP 10 FirstName, LastName FROM Person WHERE  
FirstName LIKE '%{{searchOne}}%'
```

```
SELECT TOP 10 FirstName, LastName FROM Person WHERE  
FirstName LIKE '%'; DROP TABLE TestTable; SELECT '%'
```

We were able to turn the original query into three valid commands that performed unexpected operations on the database.

# MODIFY QUERY TO USE A PERSON PICKER INSTEAD OF TEXT

```
{% assign searchTwo = 'Global' | PageParameter:'Person' | PersonByAliasGuid %}  
SELECT TOP 10 FirstName, LastName FROM Person WHERE FirstName LIKE '%  
{searchTwo.FirstName}%'
```

We're not accepting input directly from the URL so it should be safe, right?

# NAMES ARE ALSO USER INPUT



**'; DROP TABLE TempTable; SELECT ' DbTeamFamily**

Member

Protection Profile: Medium

Unknown

# A MORE SUBTLE INJECTION COMMAND

```
UPDATE UserLogin SET PersonId = 1 WHERE PersonId = 12345
```

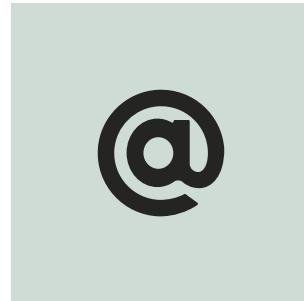
Security is tied to person records not user logins. By changing the associated person record with a newly created login we can gain unauthorized access with our own credentials

# TWO WAYS TO PREVENT SQL INJECTION



## Sanitize User Input

Escaping single quotes in user input and performing operations to make sure number inputs are in fact valid numbers



## Parameterize User Input

Send all user input as a SQL parameter instead of using Lava to generate the SQL statement

# SANITIZING USER INPUT

Sanitizing input tries to ensure the user cannot add additional commands to the SQL we intend to run

- **SanitizeSql Lava Filter**

Replaces single quotes with two single quotes so the character is escaped in the SQL query

**WHERE FirstName LIKE '{{search}}'**

If the user cannot add another single quote to end this statement and begin a new one, they cannot inject their own commands

- **AsInteger Lava Filter**

Ensures input provided is a number and not text with another command

**WHERE Id = {{personId}}**

Ensuring the input is a number and not text also protects us from unauthorized commands

# SANITIZING USER INPUT

Using our first example, we can see adding the the SanitizeSql filter keeps the user input inside the single quotes of our comparison instead of running new commands

```
{% assign searchOne = 'Global' | PageParameter:'Name' | SanitizeSql %}  
SELECT TOP 10 FirstName, LastName FROM Person WHERE FirstName LIKE '%{{searchOne}}'
```

```
SELECT TOP 10 FirstName, LastName FROM Person WHERE FirstName LIKE '%''; DROP TABLE  
TestTable; SELECT ''%
```

# PARAMETERIZING USER INPUT

To add parameters to a Lava SQL command, just add them in the opening tag

```
{% assign searchOne = 'Global' | PageParameter: 'Name' %}  
{% sql search:'{{searchOne}}' %}  
    SELECT TOP 10 * FROM Person WHERE FirstName LIKE @search  
{% endsql %}
```

# DYNAMIC DATA BLOCKS CAN MATCH SQL PARAMETERS TO URL QUERY PARAMETERS

## Query Logic

### Query i

```
1 -- {% assign searchOne = 'Global' | PageParameter:'Name' %}  
2 -- {% assign searchTwo = 'Global' | PageParameter:'Person' | PersonByAliasGuid %}  
3 SELECT TOP 10 FirstName, LastName FROM Person WHERE FirstName LIKE @name|
```

### Query is a Stored Procedure i

### Timeout

30

(sec)

### Parameters i

name=

# THE DIFFERENCE BETWEEN PARAMETERIZING AND SANITIZING

We interact with the database by sending text it interprets into commands. When the database is going to execute a set of commands it creates a plan for the operations it needs to perform.

- Sanitized input is included in the command text when the plan is made.
- The value of parameters are not included when the plan is made.

# HOW YOU CAN PROTECT YOUR SYSTEM

# PROTECT YOUR ROCK INSTANCE



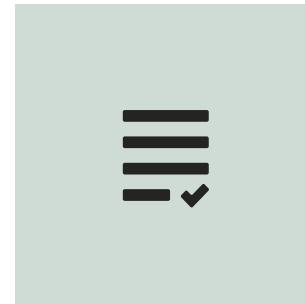
## Security Settings

- Disable duplicate checking for staff members
- Make sure staff members have elevated account protection



## Training and Policies

- Staff in charge of merging records should be trained to spot potential attackers
- Don't over-grant access to Rock



## Monitoring

- Create reports to help identify suspicious activity
- Verify staff members don't have multiple user logins
- Assign someone to check these reports regularly