

# Limiting permissions in Dynamic SQL with Execute As

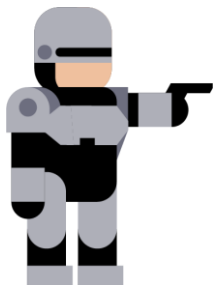
Russ Loski





# About me

- SQL Server developer 20+ years
- Focus on ETL
- Husband, father and grandfather



<https://github.com/rloski-public/Presentations/tree/main/SQLBits%202022>



<https://www.linkedin.com/in/russloski>



<https://twitter.com/sqlmovers>

<https://www.sqlmovers.com>

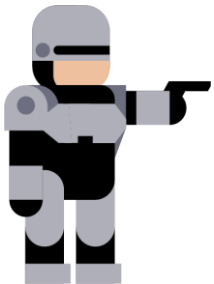
[rloski@sqlmovers.com](mailto:rloski@sqlmovers.com)





# Agenda

- Security in depth
- Admin versus limited permission
- Execute As
- Limitations



A photograph of a stone castle tower with a conical roof, partially obscured by a dark grey curved overlay. A small cartoon figure of a person with a backpack and sunglasses stands on a lower stone wall in front of the tower.

# Security in Depth

“There’s no silver bullet solution with cyber security, a layered defense is the only viable defense.”

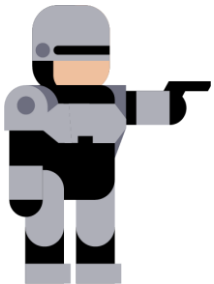
James Scott





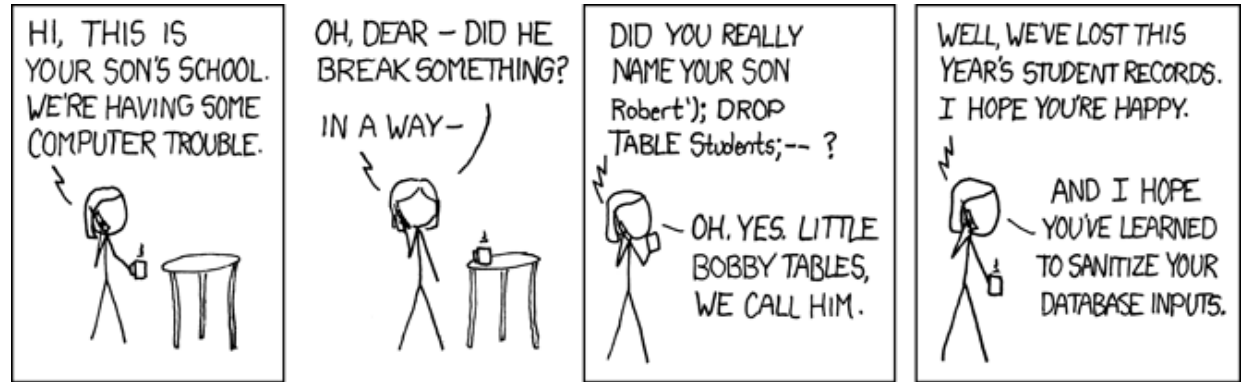
# Dynamic SQL use cases

- Admin tasks over multiple databases, tables
- Managing users
- Validation scripts
- Data load scripts in Azure data factory

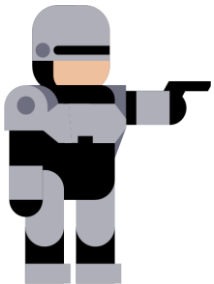


# Dangers of dynamic SQL

- SQL Injection
- Drop objects
- View protected data
- Alter data
- Add logins or users to high privilege role



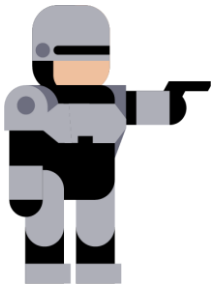
<https://xkcd.com/327>





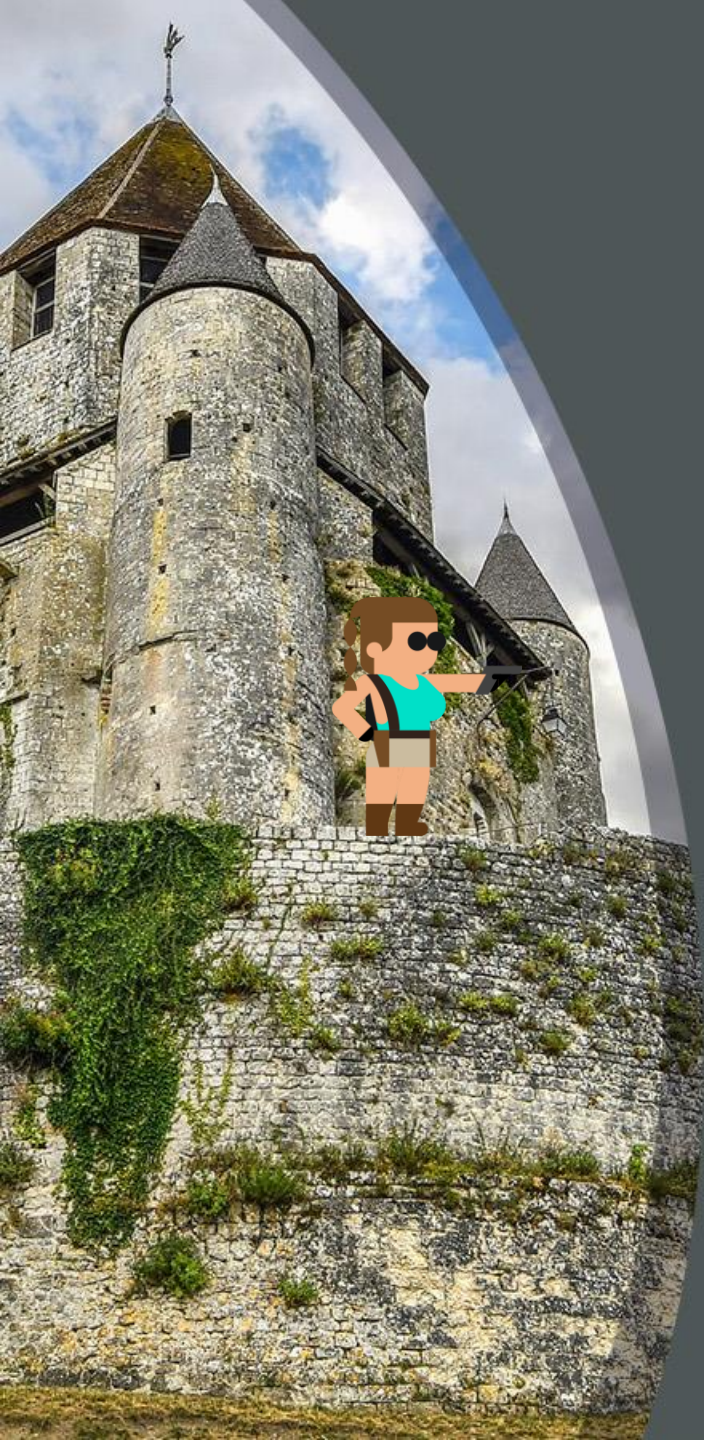
# Low permission calling user

- Session login should have just enough permission
- Session login should never be sysadmin or db\_owner



# Reduced permissions

Demo







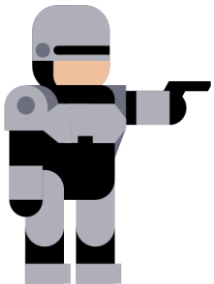
# CREATE PROC WITH Execute As

Create Procedure

@param int

With EXECUTE AS 'username'

AS



# Using execute as in procedure



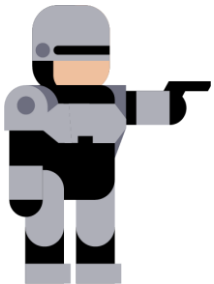
## Demo





# Limitations

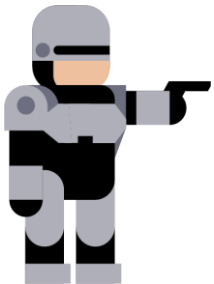
- Only user not login
- No server level permissions
- No cross database





# Caveat: Use other protections

- Don't ignore other practices
  - Scrubbing
  - QUOTENAME()

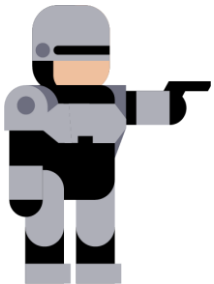






# Summary

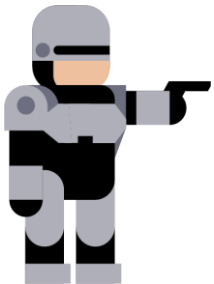
- Use caller with just enough permission
- Use “Execute As” in creating stored procedure
- Use other techniques to limit your dynamic SQL





# Resources: Castles

- [Defence in Depth: The medieval castle approach to internet security – MedStack](#)
- [The Medieval Castle's Best Defense Features & Mechanisms | by Sabana Grande | Lessons from History | Medium](#)

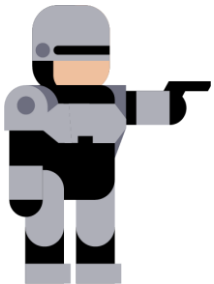






# Resources: Execute As

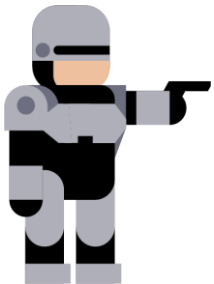
- [EXECUTE AS \(Transact-SQL\) - SQL Server | Microsoft Docs](#)
- [CREATE PROCEDURE \(Transact-SQL\) - SQL Server | Microsoft Docs](#)



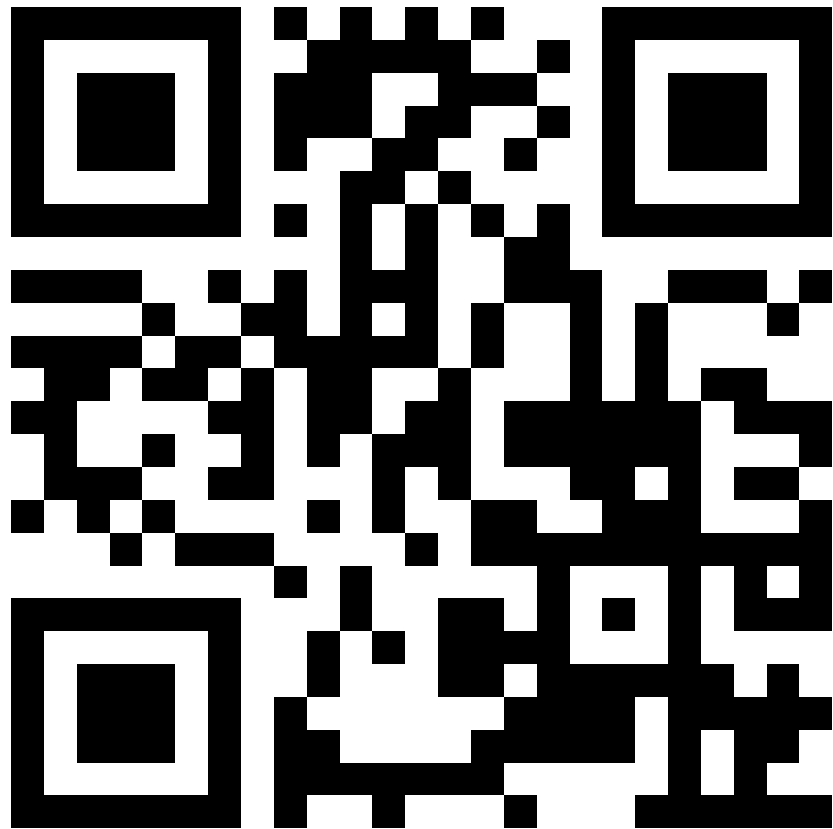


# Resources: SQL Injection

- [SQL injection cheat sheet: 8 best practices to prevent SQL injection | Snyk](#)
- [SQL Injection: Detection and prevention \(sqlshack.com\)](#)
- [SQL Injection - SQL Server | Microsoft Docs](#)



# Questions



<https://www.linkedin.com/in/russloski>



<https://twitter.com/sqlmovers>

<https://www.sqlmovers.com>

[rloski@sqlmovers.com](mailto:rloski@sqlmovers.com)

<https://github.com/rloski-public/Presentations/tree/main/SQLBits%202022>

<https://sqlb.it/?7109>

