

Scansione automatizzata di una rete LAN

da : <http://escher07.altervista.org>

Generalità

L'idea di base è quella di raccogliere in un solo strumento le seguenti funzionalità:

- scansione di un intervallo di indirizzi IP tramite Ping
- estrazione delle informazioni dalla macchina (nome, dominio etc.. e configurazione di rete)
- esportazione del prospetto ricavato in Excel

Si deve dire che per esigenze di questo tipo esistono già vari software gratuiti o quasi. Ad esempio per la scansione di un intervallo di IP con Ping e relativa esportazione dei dati in file TXT mi viene in mente IP Angry Scanner di <http://www.angryziber.com>, un programmino free utile e veloce ma che non dà altre informazioni oltre all'IP e all'hostname.

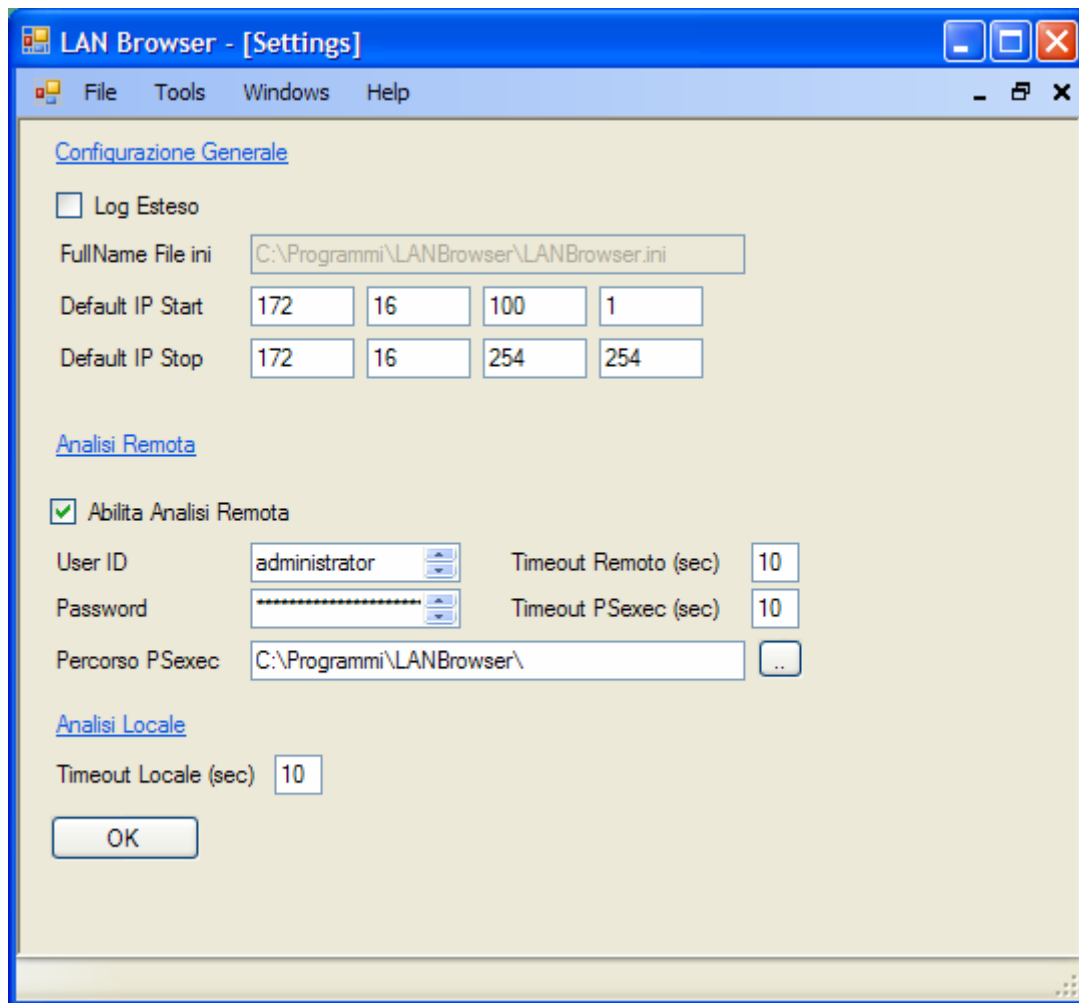
Esistono poi vari tool basati sul protocollo SNMP che svolgono lo stesso compito "sommengendoti" di informazioni ma che oltre ad essere più adatti per la scansione ed il controllo remoto di un insieme di apparati piuttosto che di una rete di PC sono spesso a pagamento.

Riguardo alla esportazione in Excel la mia esperienza mi porta a dire che è un punto essenziale perché una qualche documentazione su una rete venga mantenuta aggiornata. Ovvero le informazioni in questione devono essere aggiornate in modo automatico altrimenti si fanno delle analisi una volta ogni tanto e quando se ne ha bisogno i dati non sono mai aggiornati.

Premesso tutto questo ho sviluppato uno strumentino che dal punto di vista della eleganza non è il massimo, che in sostanza utilizza eseguibili già pronti (quali appunto utility DOS come Ping, NBTSTAT etc.. e, come vedremo PSEXEC) "affettandone" semplicemente l'output in formato testo preparando un datagrid per l'esportazione e che ha vari altri (enormi) punti di miglioramento ma che in sostanza fa tutto quello che serve.

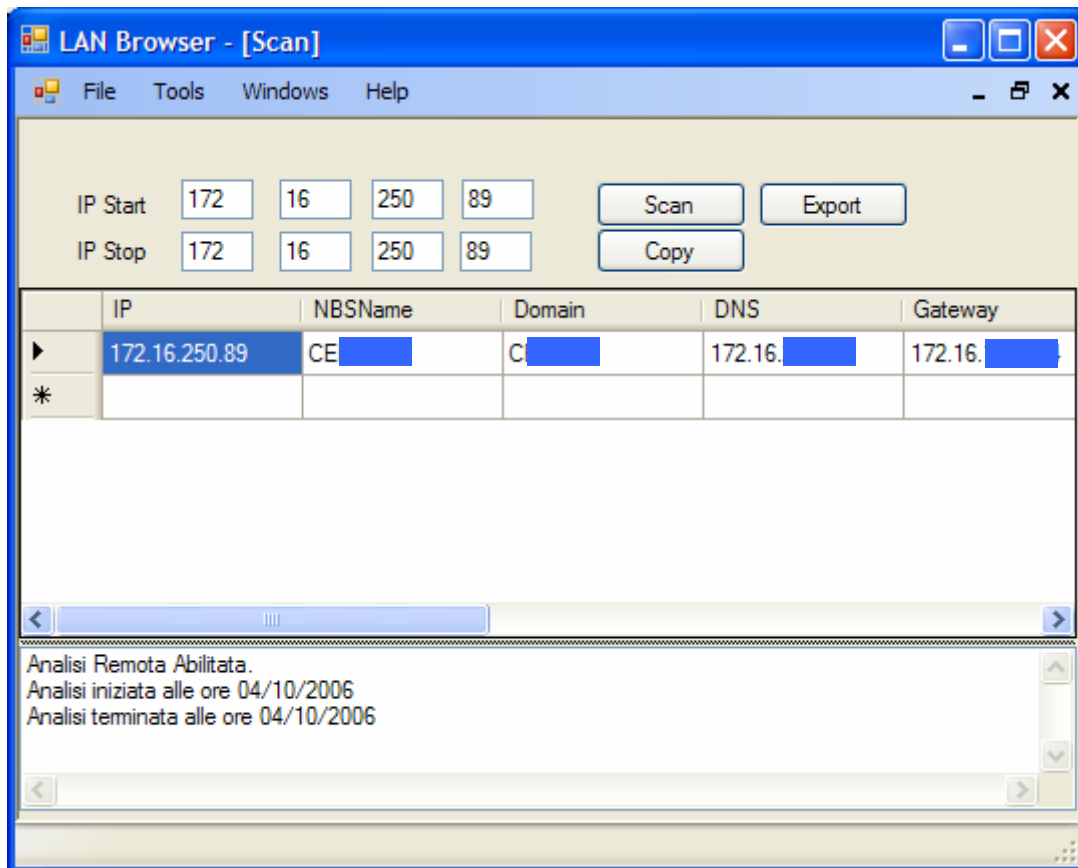
Un paio di videate chiariscono molto di più di mille altre parole, quindi qui di seguito diamo l'idea di come il tutto funziona.

Si comincia con una form ("Settings") di questo tipo:

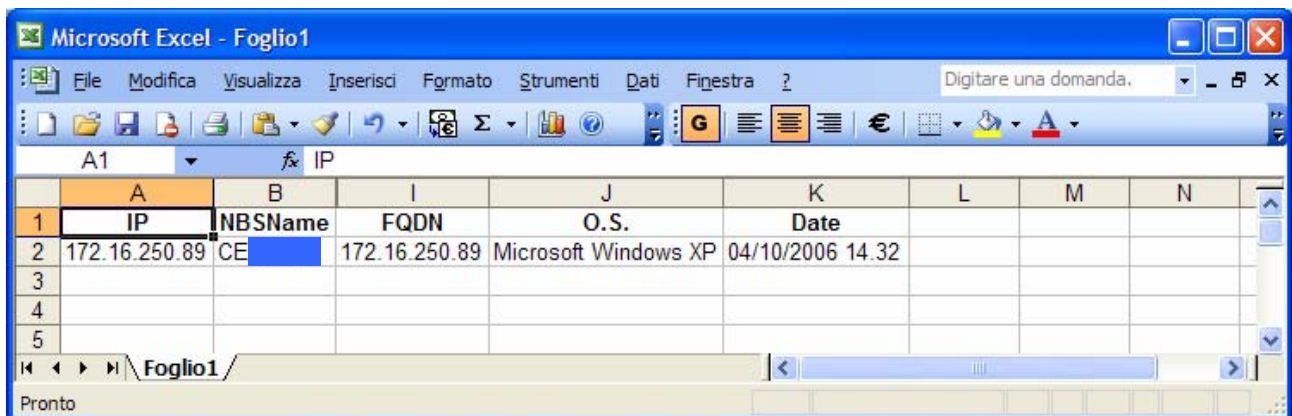


... nella quale vengono caricate le impostazioni predefinite per lo scan, contenute appunto nel file LANBrowser.ini. Per le modalità precise di utilizzo e configurazione si rimanda al successivo paragrafo "Dettagli di Utilizzo".

Si ottiene, alla fine, un qualcosa del genere (le parti in blu sono state aggiunte sullo screenshot per non compromettere la privacy della rete utilizzata per i test, non si sa mai...) :



Notiamo che utilizzando il tasto copy vengono selezionate tutte le celle del datagrid e con Incolla su un file Excel possiamo riportare il risultato.



Dettagli di utilizzo

Le informazioni contenute nella scheda [Settings] possono essere modificate e queste modifiche condizionano la successiva scansione. Le modifiche però non vengono mantenute : affinché ciò succeda bisogna editare il file ini che ha una struttura autoesplicativa come questa:

- *
 - * Configurazione Generale
- *

EnableVerboseLog=0;

```
IniPath=C:\Programmi\LANBrowser\  
IP11=172;  
IP12=16;  
IP13=100;  
IP14=1;  
IP21=172;  
IP22=16;  
IP23=254;  
IP24=254;  
*  
* Configurazione Analisi Locale  
*  
LocalTimeout=10;  
*  
* Configurazione Analisi Remota  
*  
* NB: per la password nulla usare <NULL>  
*  
EnableRemoteAnalysis=1;  
RemoteTimeout=10;  
PsExecTimeout=10;  
PsExecPath=C:\Programmi\LANBrowser\  
User=administrator;  
Password=passwordconcuiprovare_1;  
User=administrator;  
Password=passwordconcuiprovare_2;  
User=administrator;  
Password=<NULL>;
```

Sui timeout direi che la cosa principale da dire è che se troppo lunghi causano una scansione esageratamente lunga, mentre se troppo brevi può capitare che alcune informazioni non siano raccolte.

La presenza di questi timeout è dovuta ad una gestione sincrona dei processi : questi vengono lanciati ed interrotti dopo il timeout senza alcuna interazione di ritorno. Del miglioramento nella gestione dei processi parleremo meglio comunque più oltre.

Riguardo all'analisi remota c'è da dire che può essere utile non abilitarla ad esempio quando sono sufficienti le informazioni IP, NBSName, Domain, MAC-Address e FQDN ricavabili senza entrare nell'host remoto dalla semplice cache del netbios.

Se il flag EnableRemoteAnalysis è alzato viene provata la connessione con le coppie Uid/Pwd di cui nel file ini. Se con una coppia la connessione ha successo questi dati vengono scritti nel datagrid finale. Notare che per provare con la password nulla occorre utilizzare il valore speciale <NULL>.

Il tramite per eseguire la login con le credenziali di cui sopra sulla macchina remota è un programmino di terze parti ovvero PsExec di Sysinternals (<http://www.sysinternals.com>).

Questa che è un'utility da riga di comando molto ben fatta e che funziona su varie versioni di Windows e che permette (noto l'utente amministratore della macchina e la relativa password) di aprire una shell, lanciare comandi (come ad esempio un ipconfig /all) e avere a disposizione in un file di testo della macchina locale l'output del processo lanciato sulla macchina remota. Per gli scopi che ci siamo proposti i problemi sono legati al fatto che PsExec esegue il comando su una sola macchina, che non ha interfaccia grafica e che soprattutto il suo output è comunque da rielaborare cosa che non è problematica quando si deve analizzare una macchina ma lo è quando gli host sono decine o centinaia.

Nella maschera [Scan] sono presenti due bottoni utilizzabili per la esportazione : Export e Copy. Il primo utilizza la primary office interoperability di XP e, in pratica, apre Excel e vi incolla i dati. Aspetto da notare è che però se gli Office XP PIA non sono installati (correttamente) sul PC questo bottone causa un crash con la perdita di tutti i dati della scansione. Il secondo, più "modestamente" copia nella clipboard le celle selezionate (es. cliccando sulla cella a sinistra in alto si selezionano tutte, come in excel) : se a seguito di una selezione e di "copy" ci si posiziona su un foglio excel si ottiene il medesimo risultato, indipendentemente dalla presenza delle XP PIA sulla macchina.

Punti di miglioramento

Ce ne sono moltissimi ... il programma può essere ritenuto tranquillamente un semilavorato, poco più di un punto di partenza, che per un programmatore di professione risulti poco più di un accrocchio.

Se questo è innegabile (qui di seguito evidenzierò alcuni punti che mi sono sembrati i più macroscopici ma ce possono essere tranquillamente altri!) devo però altrettanto dire che è uno strumentino che rispetta pienamente le richieste iniziali.

Ho impiegato poco tempo a scriverlo (di getto, oserei dire..), fa quello che deve ed in più è open source quindi è non solo possibile ma anche facile applicarci le piccole modifiche del caso. Di fatto lo sto utilizzando da circa un anno e lo ritengo un qualcosa che mi dà un aiuto tangibile nel mio lavoro di amministratore di rete.

Detto questo veniamo alle dolenti note.

A parte considerazioni stilistiche di programma scritto in modo poco object oriented (è in C# ma lo stile è abbastanza "VB6 like") direi che i difetti principali sono questi:

- gestione dei processi esageratamente semplificata
- utilizzo di batch piuttosto che di routines di codice

Che la gestione dei processi non è il massimo lo si può vedere dall'effetto di "congelamento" durante la scansione. Inoltre c'è da dire che i processi vengono lanciati (in locale ed in remoto) senza gestire i messaggi di ritorno di fine processo per proseguire il ciclo. In pratica quindi si dà il comando (tipo "provati a connettere sulla macchina X" equivalente di Start/Esegui/Cmd PsExec [con tutti i suoi argomenti]), si aspetta un intervallo pari al timeout, si lancia il successivo e così via. Parimenti il singolo processo non è accessibile col ^C. E tutto questo, mi rendo conto, non è il massimo solo che non ho al

momento le conoscenze che mi consentano di rivedere tutto in una vera ottica asincrona e multithreading né il tempo necessario da investire in tal senso.

Riguardo alle implicazioni del secondo punto è forse utile un passo indietro : come è stato accennato in precedenza le informazioni, ad eccezione della risposta al ping (ottenuta con una applicazione banale del namespace Networking di C#), vengono ricavate non utilizzando delle API (ad esempio le WINAPI o il socket oppure quelle di .NET) ma degli eseguibili come NBTSTAT, IPCONFIG etc... Questi inviano dei caratteri sullo standard output e da questi con tecniche di manipolazione delle stringhe vengono ricavati i valori collocati nel datagrid. E' evidente che ciò non è il massimo né dell'eleganza né dell'efficienza : ho scelto questa strada (al posto ad esempio di una implementazione tutto sommato comoda con .NET Remoting) non solo per semplicità ma anche per una maggiore interoperabilità : lo sviluppo "da zero", o meglio di procedure basate su delle API implica necessariamente il legarsi ad una serie di condizioni (es. sistema operativo, presenza di runtime etc...) che la macchina host deve rispettare e che invece sono state già bypassate dagli exe in questione. Così NBTSTAT, IPCONFIG, PSEXEC etc.. funzionano sia su sistemi operativi windows tipo 98/ME che su quelli tipo 2000 e seguenti, che ci sia o no il framework di .NET e questa condizione era irrinunciabile nel caso concreto di applicazione, una rete con ancora un numero significativo di PC vecchi.

Un altro aspetto "poco bello" è quello della esportazione : abbiamo due bottoni, di cui uno (Export) manda il crash il programma nella maggior parte dei casi (quanti sono i PC che a tutt'oggi hanno installate le PIA?). Me ne rendo conto : la soluzione poteva, per il "Profilo alto" prevedere un controllo di presenza delle PIA nella gestione dell'evento OnClick del bottone Esporta, oppure per il profilo basso non far comparire nemmeno tale bottone. Ho preferito lasciare comunque il bottone che è comunque una funzionalità in più (e perché sul mio PC le PIA ci sono ...), anche se mi rendo conto che è appunto un punto di miglioramento aperto.

Osservazioni Varie

L'output principale è come si è detto il datagrid esportabile in Excel : va però osservato che le tutte le informazioni che vengono raccolte e poi filtrate/selezionate per diventare campi delle righe del datagrid sono lo stesso sul PC locale nella cartella di installazione del programma. Può essere utile a volte esaminarle, ad esempio per vedere gli utenti definiti sull'host remoto al di là di quello con cui si è provata la connessione.

Il proxy, come da documentazione Microsoft è scritto nel registro di configurazione ed è un attributo dell'utente, per cui se fate la connessione con Administrator e l'utente normalmente utilizzato sulla macchina è Tizio è molto probabile (a meno che le impostazioni LAN di IE non siano state configurate anche per l'utente amministratore, cosa che personalmente non faccio quasi mai) che il corrispondente campo nel risultato della scansione possa essere nullo.

Implicazioni di Sicurezza

Il programma nasce come tool di amministrazione di rete. Personalmente io l'ho trovato utile per monitorare soprattutto reti con workgroup ed in particolare per :

- controllare la corretta assegnazione dei gateway e DNS
- verificare i gruppi e la loro composizione
- controllare (in una rete a workgroup) quali sono le macchine che non hanno la password di amministratore che dovrebbero avere
- fare un inventario della rete prima di migrazioni a strutture con dominio

Inoltre i suoi tabulati, visto che riportano anche la data della scansione possono essere utilizzati (ad esempio importandoli in un DB tipo access) per verificare la storia delle modifiche sui vari nodi. Il programma può essere poi impiegato insieme a NetController (di cui in questo sito) per preparare la lista dei nodi da sottoporre a monitoraggio continuo e così via.

Questi sono gli usi per così dire "leciti" : se la rete è "vostra" in pratica il programma non fa niente di diverso da quello che avreste dovuto fare manualmente : connettervi ad ogni PC con la "vostra" password di amministratore e raccogliere le informazioni che vi occorrono per la gestione della rete stessa.

Ciò assodato è senz'altro utile, però, tenere presenti i seguenti fatti:

- LANBrowser esegue una scansione e lo fa su qualunque rete richiediate;
- se l'analisi remota è abilitata di fatto esegue una sequenza di accessi col metodo bruteforce a dizionario;
- se avete un minimo di competenze è possibile che i batch eseguiti in remoto diano comandi ben più pericolosi di un ipconfig /all.

Questi aspetti ci fanno capire che anche se non nasce come tool di hacking un_suo uso scorretto potrebbe portare ai medesimi effetti dell'attacco di un hacker malevolo.. Del resto questo è vero per ogni tool di amministrazione remota ed è pur vero che il programma non fa niente che di per sé non rientra nelle funzionalità base di Windows, per cui in sintesi affinché possa portare ad azioni che impichino responsabilità civile o penale è necessaria una componente di dolo o colpa grave da parte di chi lo utilizza, che sono ovviamente elementi esterni a questo software.

Disclaimer

Ricordo che questo programma è fornito "as is" : non sono in alcun modo responsabile di tutto quello che possa derivare né dal suo uso corretto né scorretto. Ricordo parimenti che lanciare scansioni non autorizzate o tentare connessioni a PC di terzi che non hanno preventivamente acconsentito al trattamento è una violazione che può comportare responsabilità civili o penali. Potete modificare il programma come volete, anzi vi sarei grato che mi metteste a conoscenza di eventuali modifiche o sviluppi che ci avete fatto sopra per migliorarlo o adattarlo meglio alle vostre esigenze.