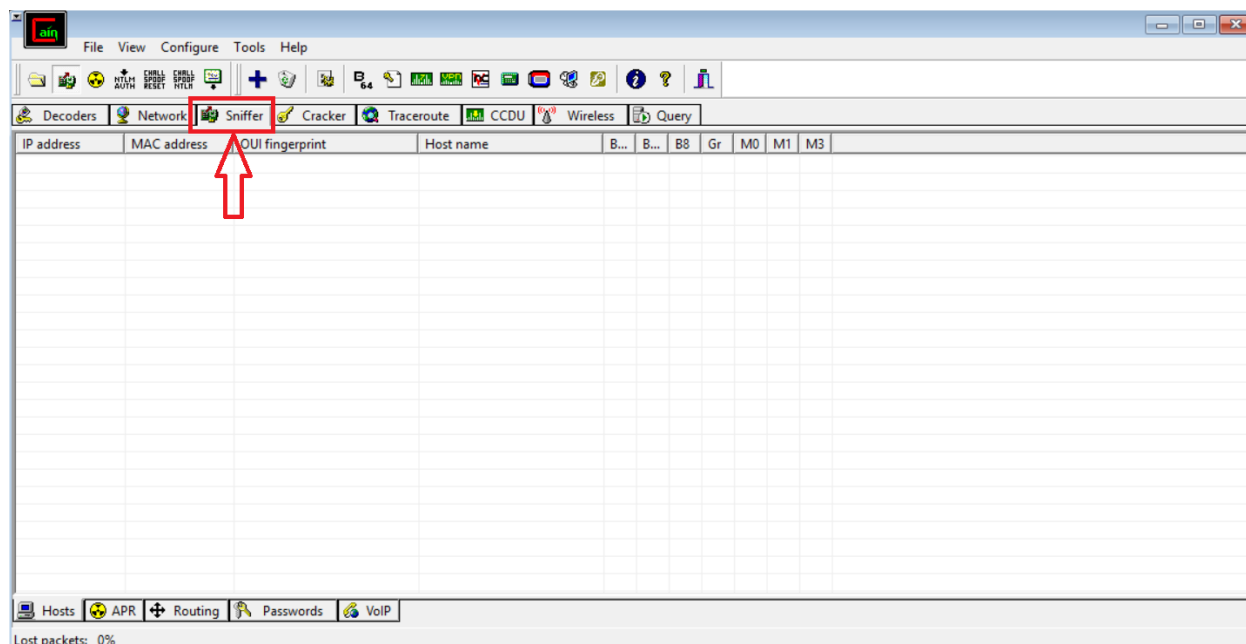
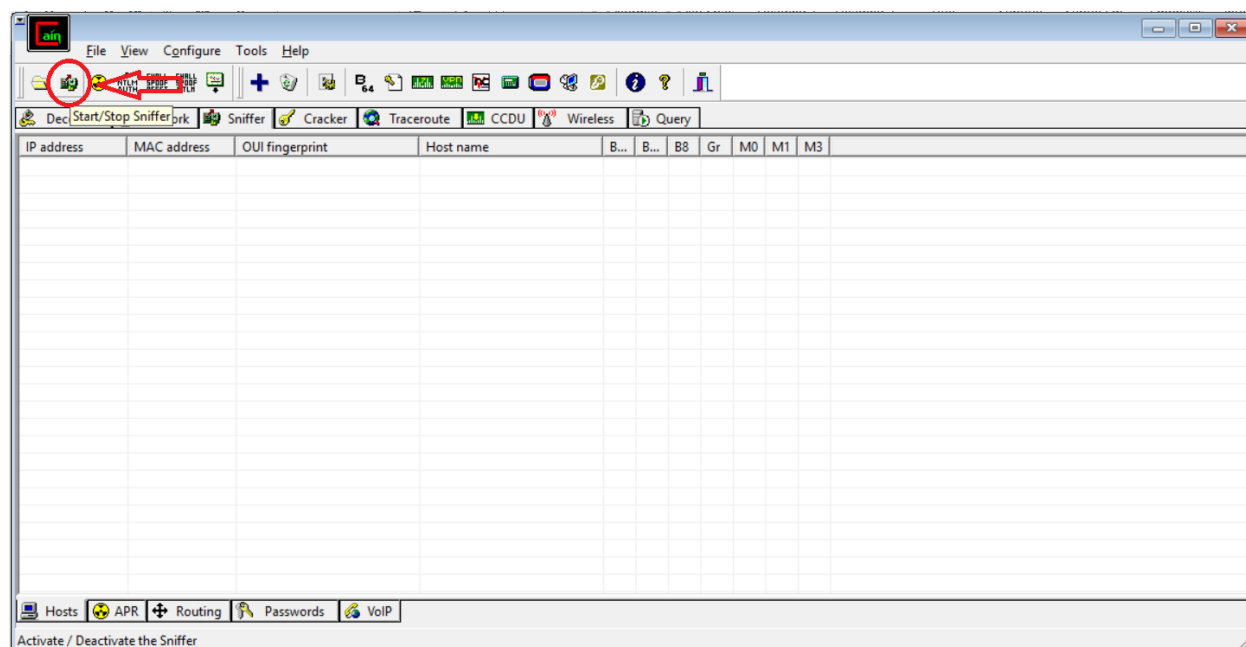


آماده سازی Cain & Abel

در ابتدا نوار مربوط به Sniffer را انتخاب می کنیم:



سپس Sniffer را روشن می کنیم:

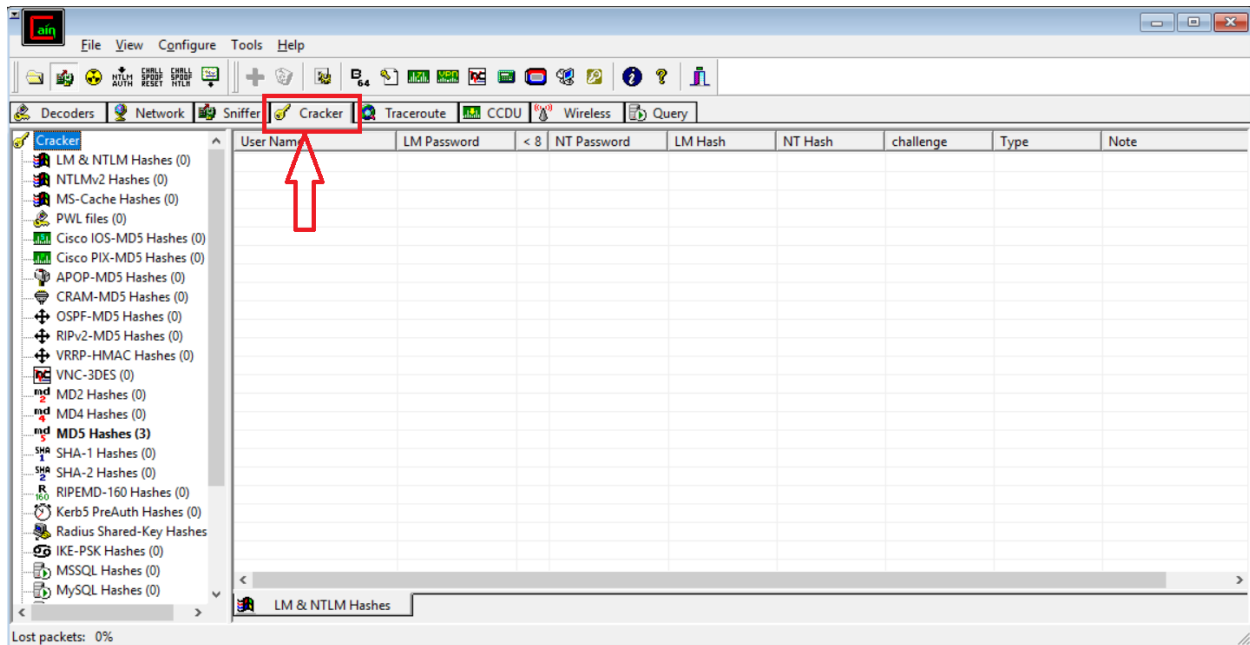


The screenshot shows the main window of Cain & Abel. The menu bar includes File, View, Configure, Tools, and Help. Below it is a toolbar with various icons for file operations and network analysis. A secondary toolbar contains tabs for Decoders, Network, Sniffer, Cracker, Traceroute, CCDU, Wireless, and Query. The 'Sniffer' tab is active. At the bottom, there is a table displaying captured network traffic:

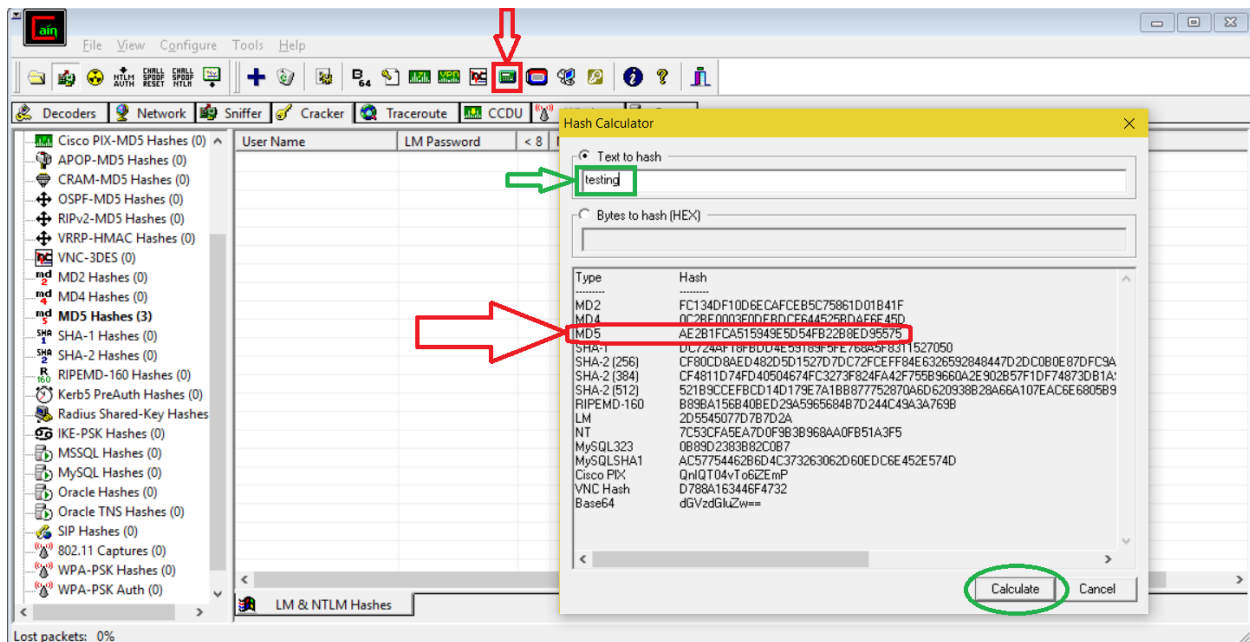
IP address	MAC address	OUI fingerprint	Host name	B...	B...	B8	Gr	M0	M1	M3
192.168.1.1	B06[redacted]DF4									
192.168.1.5	[redacted]01BDD									

A red arrow points to the OUI fingerprint column header.

سپس نوار **Cracker** را انتخاب می کنیم:

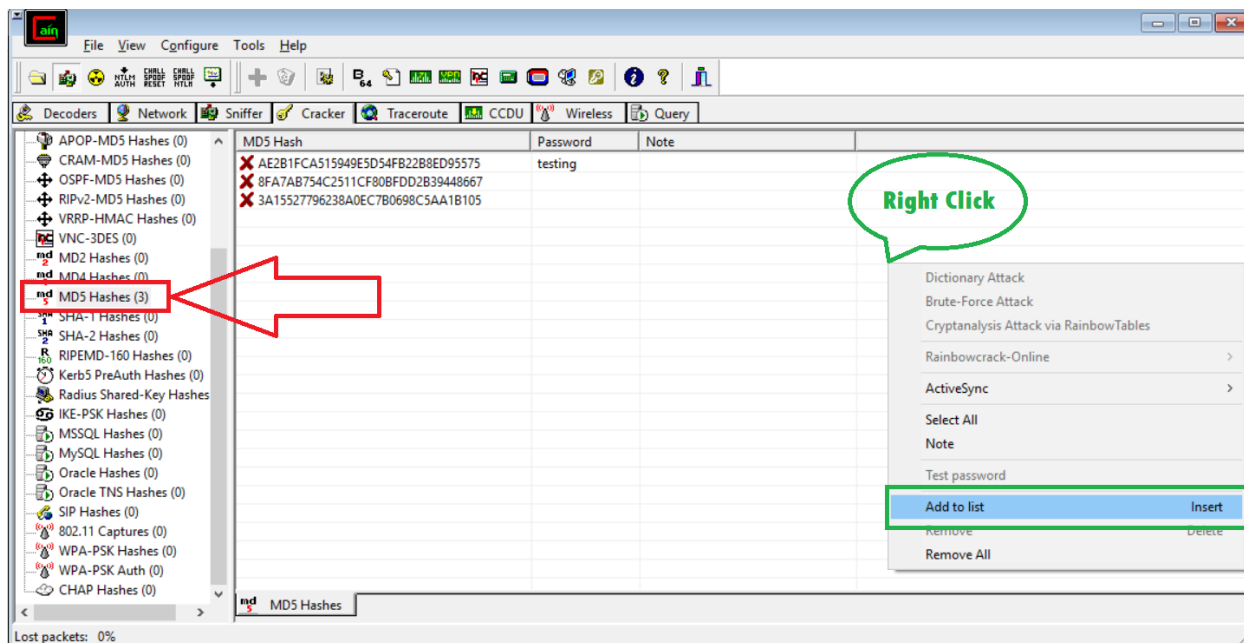


سپس برای ارزیابی و تست عملکرد، مقدار **Hash** شده‌ی گذروژه‌های نمونه را به آن اضافه می کنیم. برای این کار می توان از ابزار **HASH CALCULATOR** تعبیه شده در **Cain & Abel** استفاده کرد:

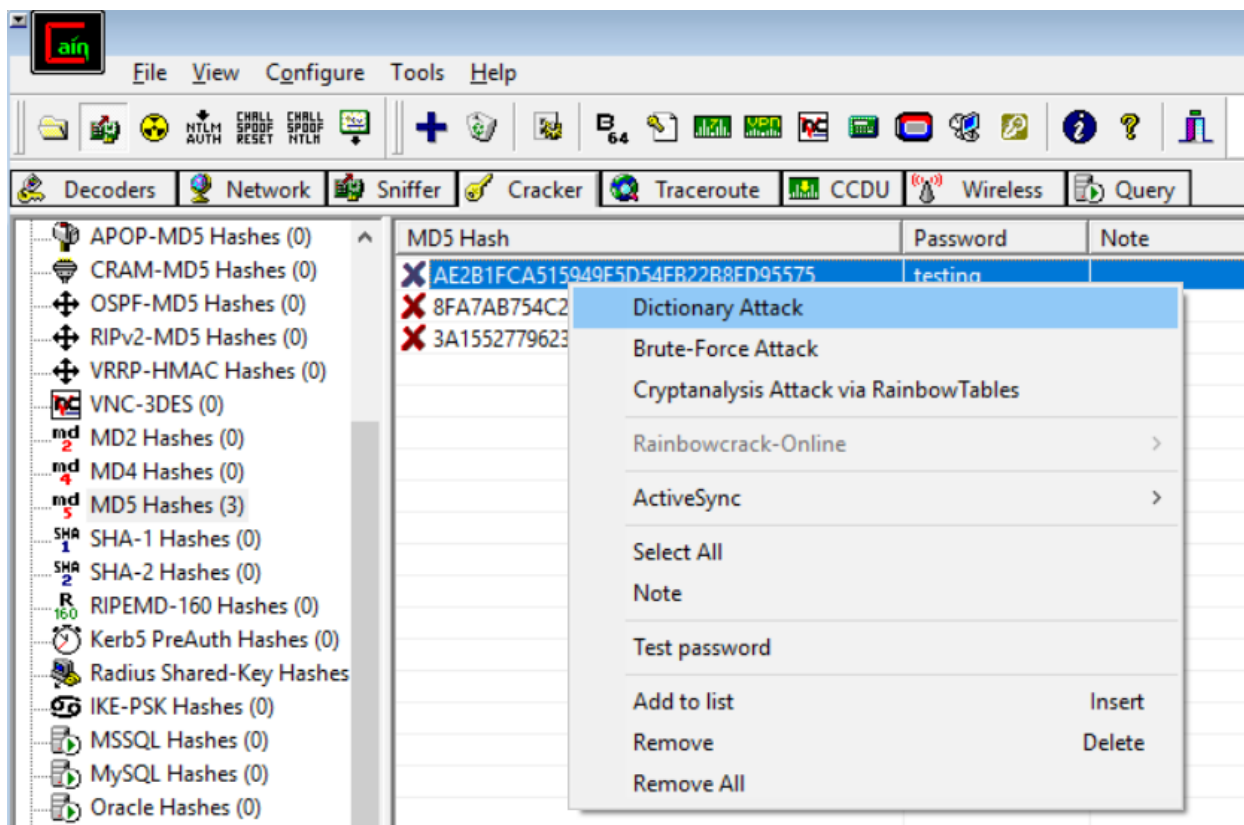


برای سادگی کار و سریع تر شدن محاسبه‌ی Hash ها در ادامگی کار، از **MD5** استفاده می کنیم.

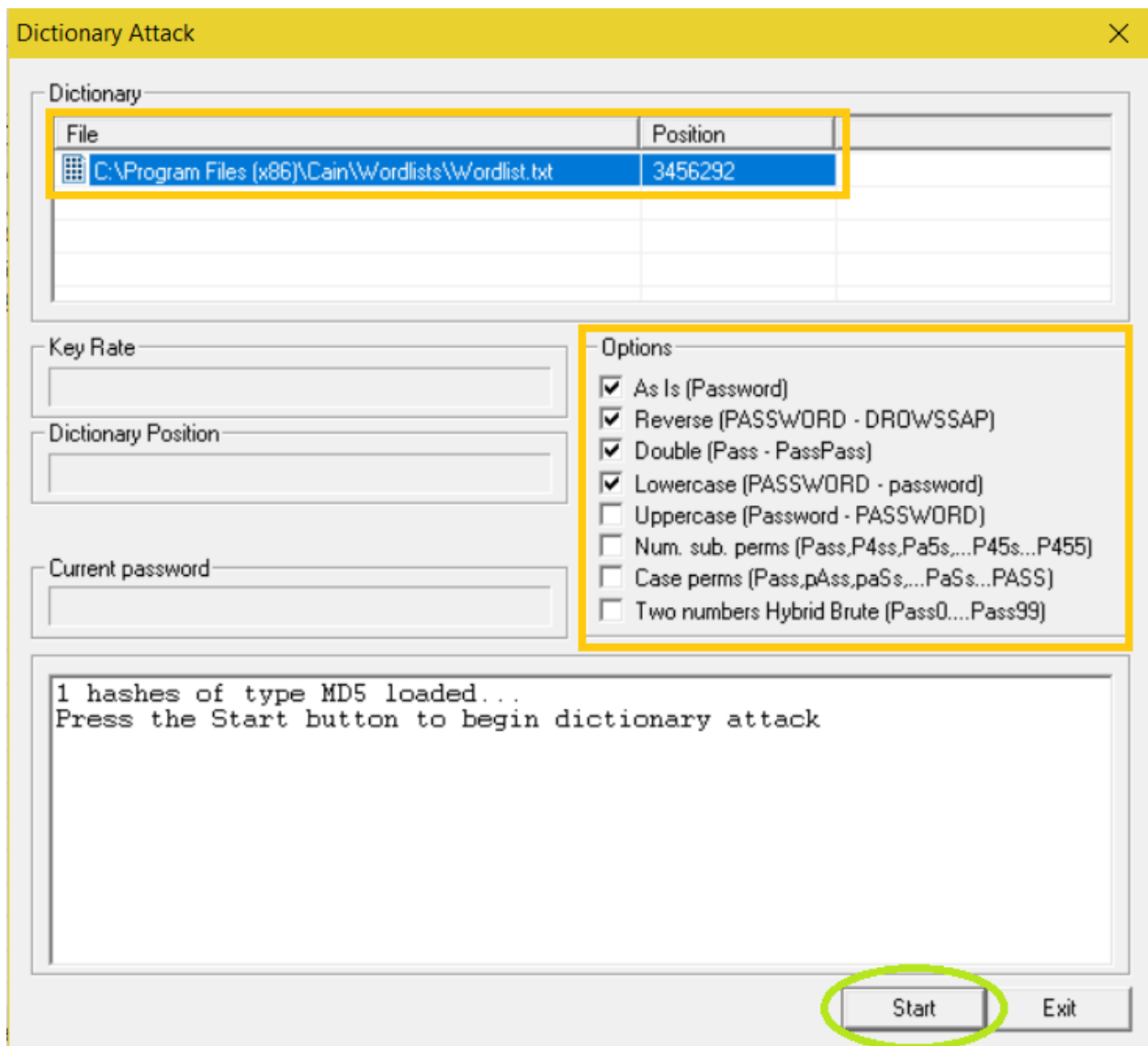
سپس از نوار سمت چپ، گزینه‌ی MD5 Hashes را انتخاب می‌کنیم و مقادیر مورد نظر را به آن اضافه می‌کنیم:



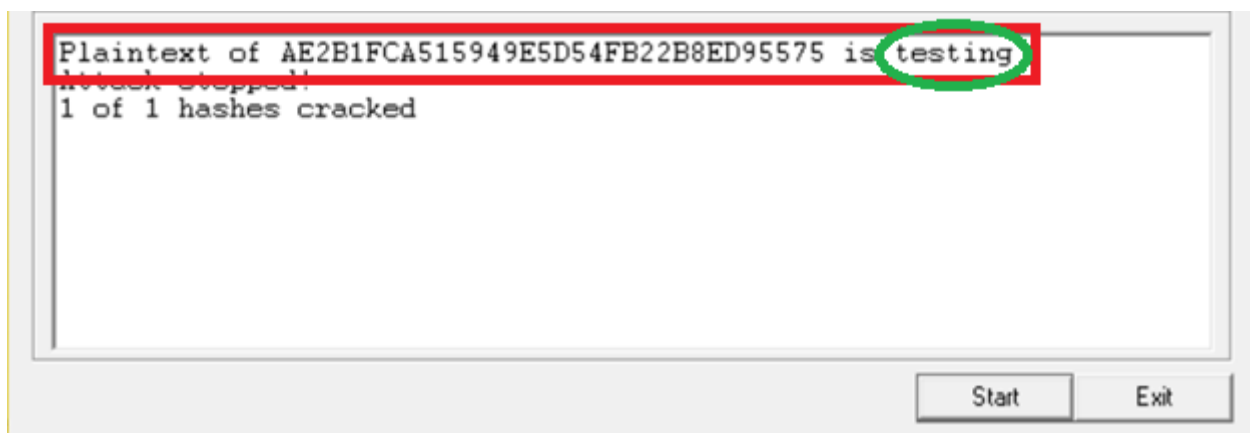
حال انواع حملات را انجام می‌دهیم:



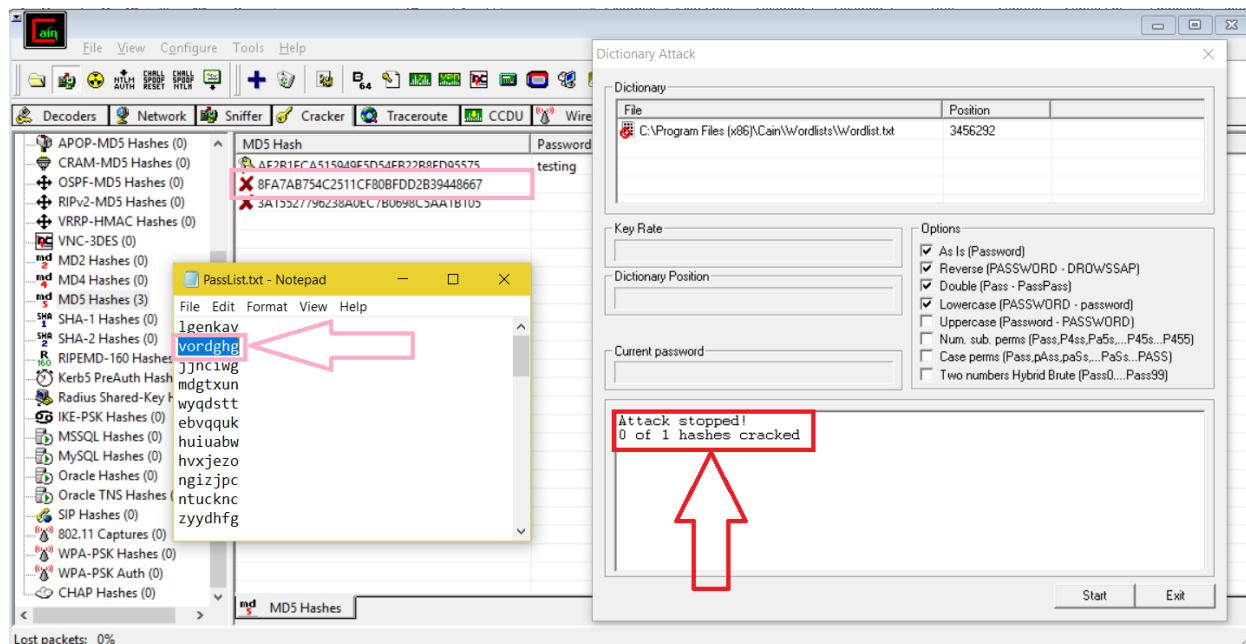
تنظیمات مربوطه را انجام می‌دهیم:



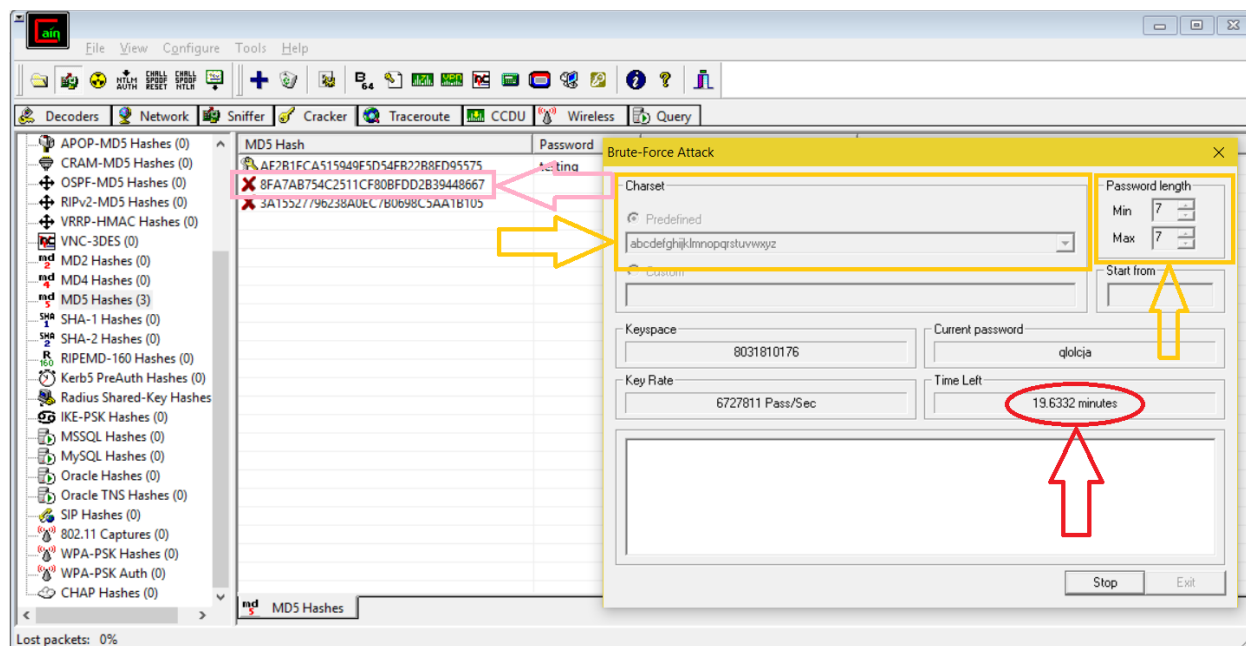
سپس نتیجه نمایان می‌شود:



پس به این ترتیب دیدیم که اگر گذرواژه‌ها ساده و از کلمات معنادار و پرتکرار باشند، به سرعت توسط حمله‌ی Dictionary قادر به پیدا کردن آن‌ها خواهیم بود؛ اما در مثال بعدی، می‌بینیم که این نوع از حمله، برای گذرواژه‌ی تصادفی و غیرمعقول و یا به طور کلی، غیر شفاف، نمی‌تواند نتایج‌ای بیابد:



پس به سراغ گزینه‌ی بعدی می‌رویم و از حمله‌ی Brute-Force استفاده می‌کنیم:



می‌بینیم که زمان قابل توجهی نیاز است تا به نتیجه برسیم اما این زمان همچنان معقول به نظر می‌رسد.

نتیجه پس از حدود ۲ دقیقه به دست می‌آید:

Brute-Force Attack

Charset

☒ Predefined

abcdefghijklmnopqrstuvwxyz

☐ Custom

Password length

Min 7

Max 7

Start from

lgenkav

Keyspace

1539777761

Current password

Key Rate

Time Left

Plaintext of 8FA7AB754C2511CF80BFDD2B39448667 is lgenkav

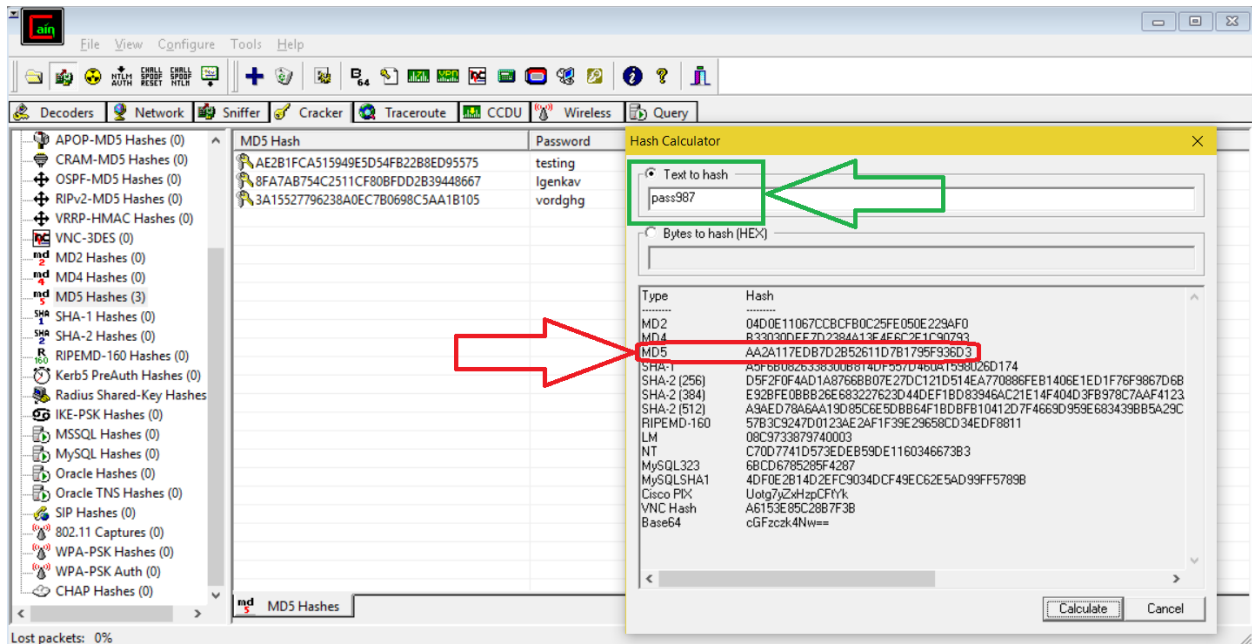
Attack stopped!

1 of 1 hashes cracked

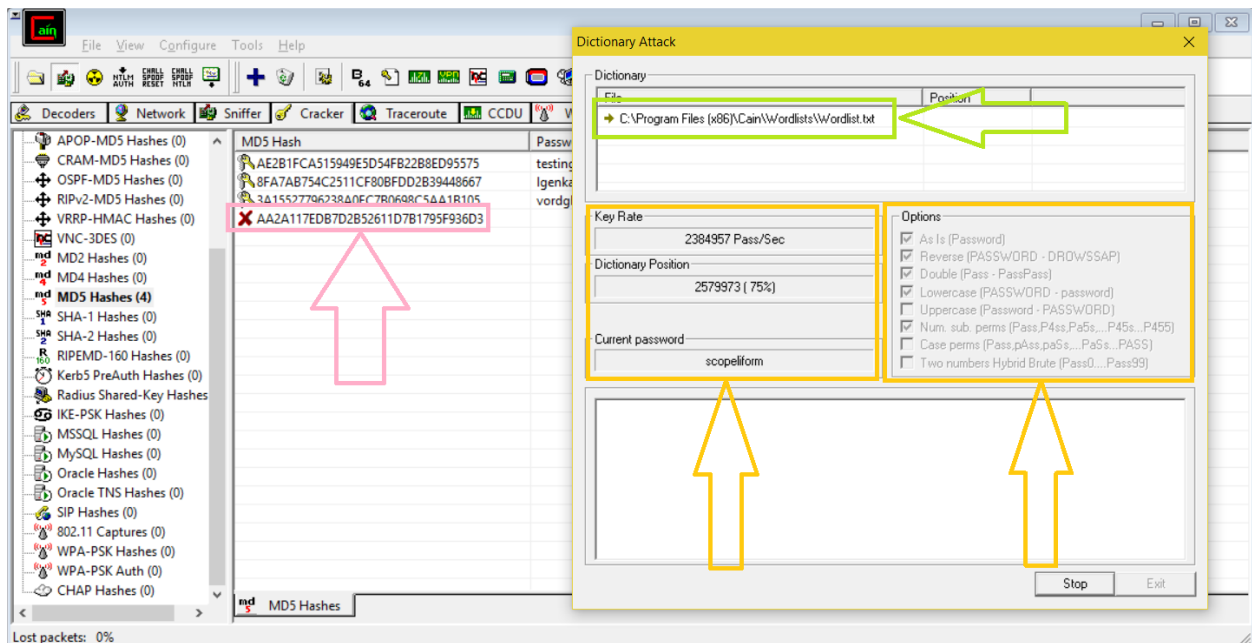
Start Exit

همین فرایندها را برای دسته‌ی دوم و سوم گذرواژه‌های مطرح شده اجرا می‌کنیم.

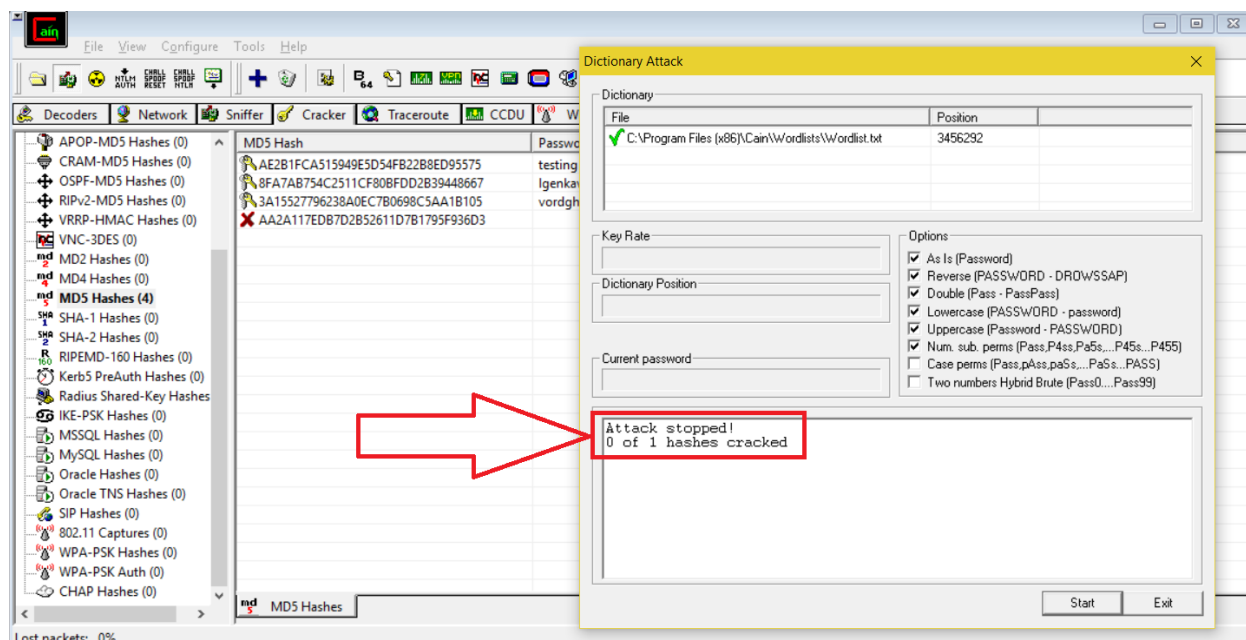
ابتدا برای گذرواژه‌های ساده داریم:



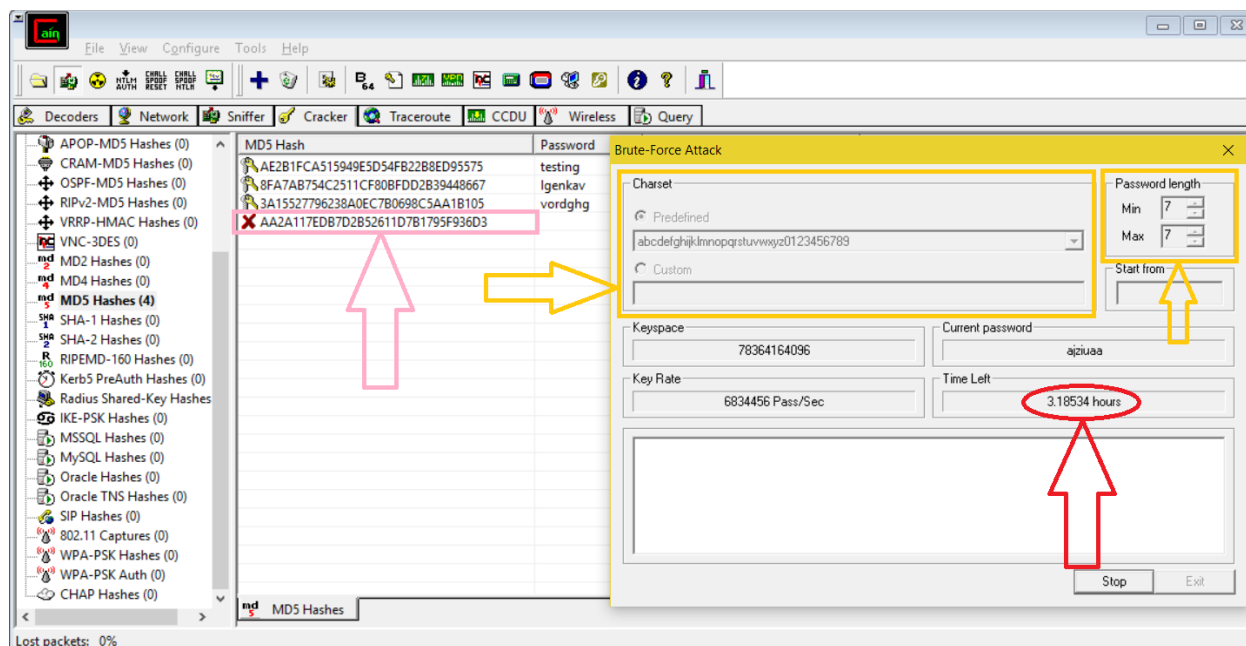
حمله‌ی Dictionary را اجرا می‌کنیم:



با استفاده از حمله‌ی Dictionary مشاهده می‌شود که حتی برای یک گذرواژه‌ی ساده که فقط عدد به آن اضافه کردیم، به نتیجه‌ای نمی‌رسیم و این گواه بر آن است که با اضافه کردن اعداد به گذرواژه‌ها، امنیت آن‌ها به مقدار قابل توجهی افزایش پیدا می‌کند:

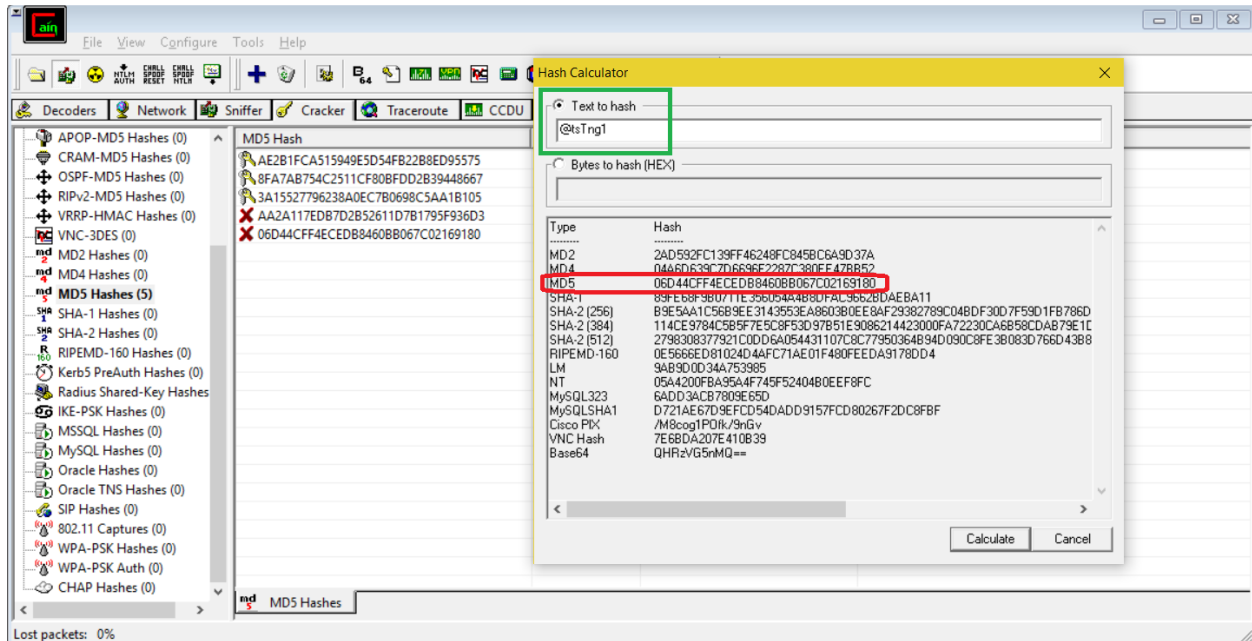


پس به سراغ گزینه‌ی بعدی می‌رویم و از حمله‌ی Brute-Force استفاده می‌کنیم:

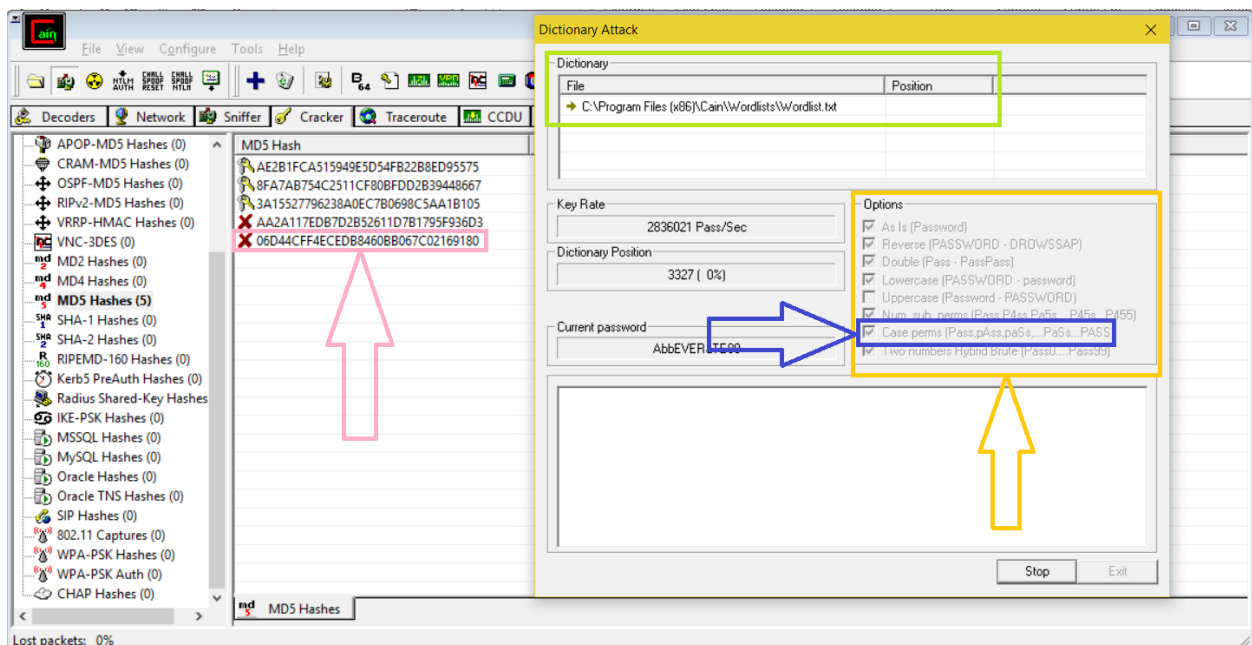


همان‌طور مشاهده می‌شود، زمان مورد نیاز برای آن که به نتیجه برسیم، تقریباً ۱۰ برابر شده است!

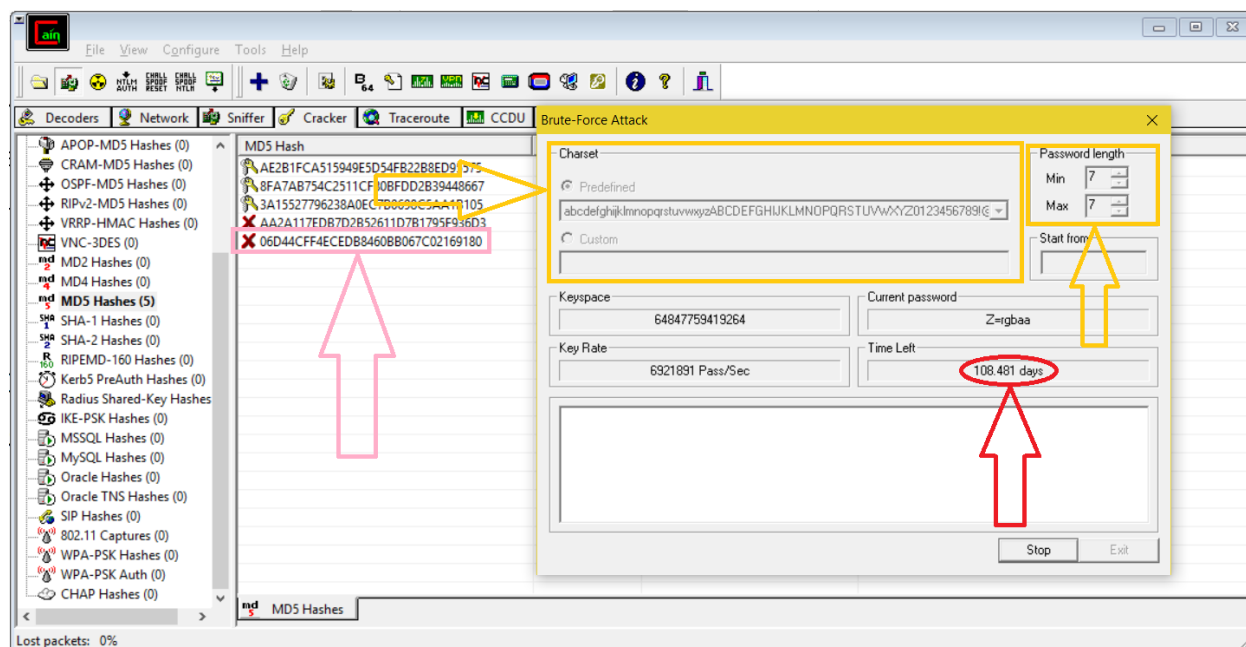
ابتدا برای گذرواژه‌های ساده داریم:



به دلیل استفاده از ترکیب حروف بزرگ و کوچک، مشاهده می‌شود که حتی حملی Dictionary نیز به زمان قابل توجهی نیاز دارد تا به نتیجه برسد:



با توجه به نتایج فوق، می‌توان نتیجه گرفت که حمله‌ی Brute-Force نیز در زمانی قابل قبول قادر نیست به نتیجه‌ی مطلوب برسد اما برای اثبات فرضیه‌مان، این نوع حمله را نیز اجرا می‌کنیم:



همان‌طور که مشاهده می‌شود، زمان لازم برای به نتیجه رسیدن، بسیار زیاد است و در عمل، استفاده از این روش منطقی به نظر نمی‌رسد.

با توجه به نتایج این دو روش و مقایسه‌ای که میان آن دو انجام دادیم، نتیجه می‌گیریم که روش Dictionary که در آن گذرواژه‌ها با استفاده از یک لیست از گذرواژه‌های رایج مورد بررسی قرار می‌گیرند، در زمان کمتری نتیجه می‌دهد ولی لزوماً به نتیجه نمی‌رسد اما روش Brute-Force که در آن تمام گذرواژه‌های ممکن با استفاده از کاراکترهای معین بررسی می‌شود، حتماً به نتیجه خواهد رسید اما زمان بسیار زیادی نیاز دارد که گاهی اوقات عملاً نشدنی به نظر می‌رسد اما در تئوری قطعاً نتیجه‌بخش است. پس نمی‌توان مطلقاً نظر داد و گفت کدام مؤثرتر است؛ زیرا هر کدام مزایا و معایبی دارند.

اجرای برنامه‌ی winrtgen.exe

به طور کلی، Rainbow Tableها، جدول‌هایی شامل مقادیر Hash شده‌ی از قبل محاسبه شده است. حال با استفاده از برنامه‌ی فوق، یک تحلیل مختصر ارائه می‌دهیم:

The screenshot shows the 'Rainbow Table properties' dialog box. It contains several sections and fields:

- Hash:** A dropdown menu set to 'md5'.
- Min Len:** A text box containing '7'.
- Max Len:** A text box containing '7'.
- Index:** A text box containing '0'.
- Chain Len:** A text box containing '2400'.
- Chain Count:** A text box containing '40000000'.
- N° of tables:** A text box containing '1'.
- Charset:** A dropdown menu set to 'all-space'. To its right is an 'Edit' button.
- Table properties:** A section containing:
 - Key space: 7446353252589 keys
 - Disk space: 610,35 MB
 - Success probability: 0.012763 (1.28%)
- Benchmark:** A section containing:
 - Hash speed: 6172839 hash/sec
 - Step speed: 3242542 step/sec
 - Table precomputation time: 8.224 hours
 - Total precomputation time: 8.224 hours
 - Max cryptanalysis time: 0.888192 seconds
- Optional parameter:** A text box containing 'Administrator'.
- Buttons:** 'Benchmark', 'OK', and 'Cancel'.

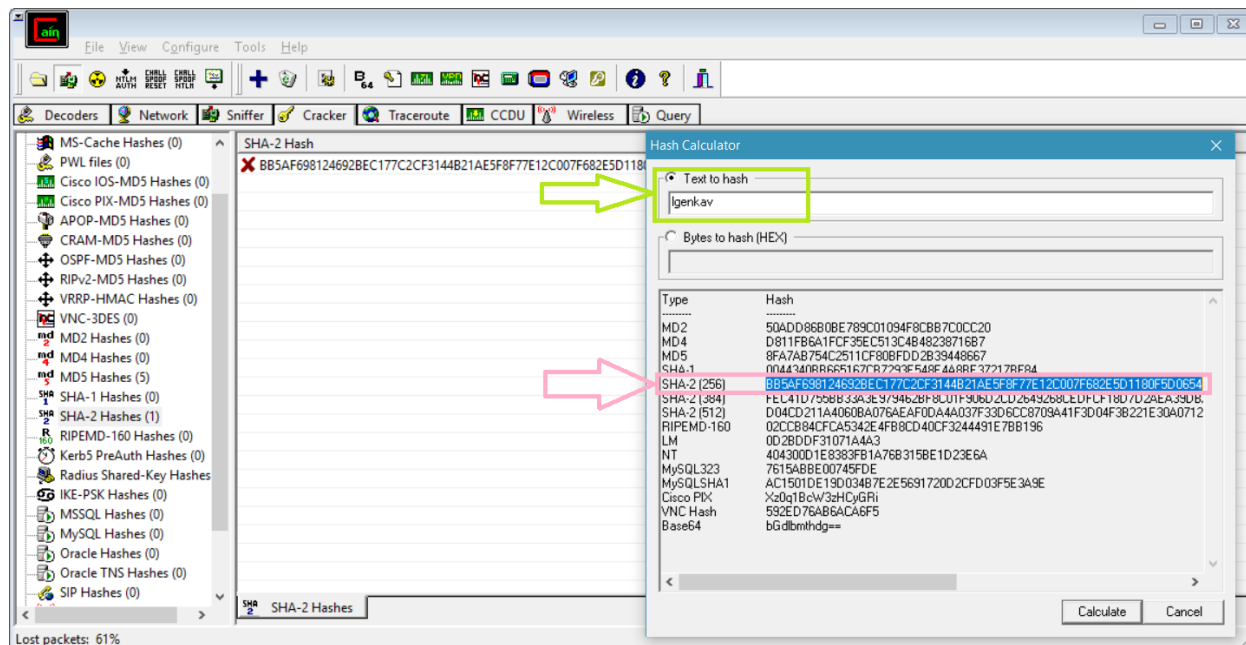
Colored boxes and arrows highlight specific areas:

- A yellow box highlights the 'Hash', 'Min Len', and 'Max Len' fields.
- A pink box highlights the 'Charset' dropdown and the 'Edit' button.
- A green box highlights the 'Table properties' section.
- A red box highlights the 'Benchmark' section.
- A red arrow points from the 'Optional parameter' text box to the 'Benchmark' section.

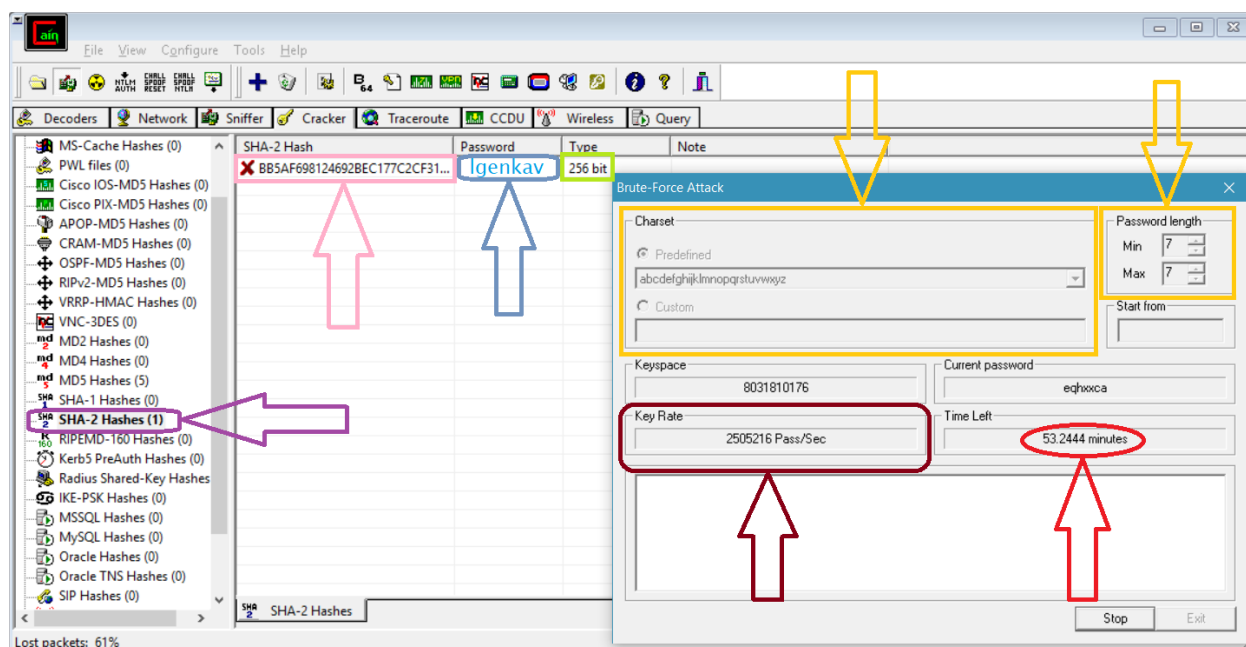
همان‌طور که مشاهده می‌شود، برای دسته‌ی سوم گذرواژه‌ها با طول ۷ کاراکتر، به حدود ۸ ساعت زمان برای ساخت این جدول زمان لازم است و حافظه‌ی زیادی نیز می‌طلبد اما در مقابل، حداکثر زمان لازم برای Cryptanalysis کمتر از یک ثانیه خواهد بود؛ به عبارتی با یک‌بار محاسبه‌ی مقادیر Hash، می‌توان صرفاً به مقایسه‌ی این مقادیر پرداخت و دیگر نیازی به محاسبه‌ی مجدد نیست. به عبارتی دیگر، یک‌بار محاسبات را انجام می‌دهیم و نتیجه را ذخیره می‌کنیم تا در آینده از آن استفاده کنیم و نیازی به صرف زمان مجدد نداشته باشیم.

الگوریتم‌های ذخیره‌سازی مختلف

همان‌طور که قبلاً بیان شد، ما از MD5 Hash برای ذخیره‌سازی گذرواژه‌ها استفاده کردیم. حال از SHA-2 Hash نسخه‌ی ۲۵۶ بیتی استفاده می‌کنیم:



سپس، برای همان گذرواژه‌ای که قبلاً بررسی کردیم، حمله‌ی Brute-Force را دوباره اجرا می‌کنیم:



همان‌طور که مشاهده می‌شود سرعت محاسبه‌ی مقادیر SHA-2 Hash برای گذرواژه‌های یکسان، کمتر از نصف سرعت محاسبه‌ی مقادیر MD5 Hash همان گذرواژه‌هاست و به همین نسبت نیز زمان بیشتری مورد نیاز است تا گذرواژه‌های ذخیره شده با الگوریتم SHA-2 را بررسی کنیم و به نتیجه برسیم. به نظر می‌رسد سرعت محاسبه‌ی MD5 Hash بر روی سخت‌افزار یکسان، بیشتر از انواع دیگر توابع Hash است.

آیا Cain & Abel بدافزار است؟

بسیاری از آنتی‌ویروس‌ها و حتی مرورگر *GOOGLE CHROME* آن را بدافزار تشخیص می‌دهند اما در حقیقت می‌توان گفت که این ابزار، مانند یک چاقو است که استفاده‌های مختلفی دارد؛ می‌توان با آن آسیب‌پذیری‌های یک شبکه‌ی داخلی را مورد ارزیابی قرار داد و ایرادات آن را برطرف کرد یا می‌توان از آن برای دسترسی و شنود غیرمجاز استفاده کرد. بنابراین در حقیقت، این که چه کسی کاربر نرم‌افزار است و هدف او از استفاده از آن تعیین‌کننده است.

اما فارغ از بحث‌های اخلاقی، می‌توان چنین گفت که این نرم‌افزار رفتارهایی نشان می‌دهد که معمولاً مطلوب کاربران عادی نیستند و از این جهت می‌توان آن را یک بدافزار قلمداد کرد و رفتارهای آن برای کاربران عادی را خطرناک دانست.

اما خود این برنامه، ماهیت یک بدافزار را ندارد و صرفاً یک ابزار کاربردی در زمینه‌ی امنیت اطلاعات است اما آنتی‌ویروس‌ها با در نظر گرفتن این که اکثر کاربران آن‌ها، کاربران عادی هستند و کارهایی از قبیل بررسی آسیب‌پذیری یک شبکه را انجام نمی‌دهند، به جهت رعایت احتیاط، آن را بدافزار قلمداد می‌کنند؛ زیرا ممکن است باعث اختلال در کارهای معمول کاربران شود و عملکرد کل سیستم را تحت تأثیر قرار دهد. به عنوان مثال، این نرم‌افزار به هیچ عنوان نباید توسط یک کارمند عادی بانک، روی یک کامپیوتر اجرا شود؛ زیرا تمام شبکه‌ی داخلی بانک را تحت تأثیر قرار می‌دهد و ممکن است اطلاعات مهم مالی از دست بروند. پس فقط متخصصان امنیت و شبکه باید بتوانند از آن استفاده کنند.