ARTeam eZine

0F 85 AC 34 FF FF 8B 85 54 FD FF FF 8B KNOWLEDGE 40 04 8B 8D 48 FD FF FF 0F B7 0C 08 8D 3F 0F B7 0C 07 8D 34 48 66 83 3E 00 0F 85 AC 34 FF FF 8B 85 54 FD FF FF 8B 40 04 8B 8D 48 FD FF FF 0F B7 DISCOVERY 0C 08 8D 34 48 E9 91 34 FF FF 8B 85 40 FB FF FF 8B 40 04 0F B7 48 32 E9 75 9F FE FF E9 11 3B FF FF 8B 7D 10 85 FF 0F 84 5D FF FF REVERSING FF 8B 75 08 8B 46 58 85 C0 0F 85 41 FF FF FF 8B 4C 85 D2 74 0B 8B 4E 50 85 C9 0F 85 EB 6A 00 00 8B 45 14 C7 07 A4 14 81 7C C7 00 02 00 00 00 E9 C8 C7 FE FF

HTTP://CRACKING.ACCESSROOT.COM

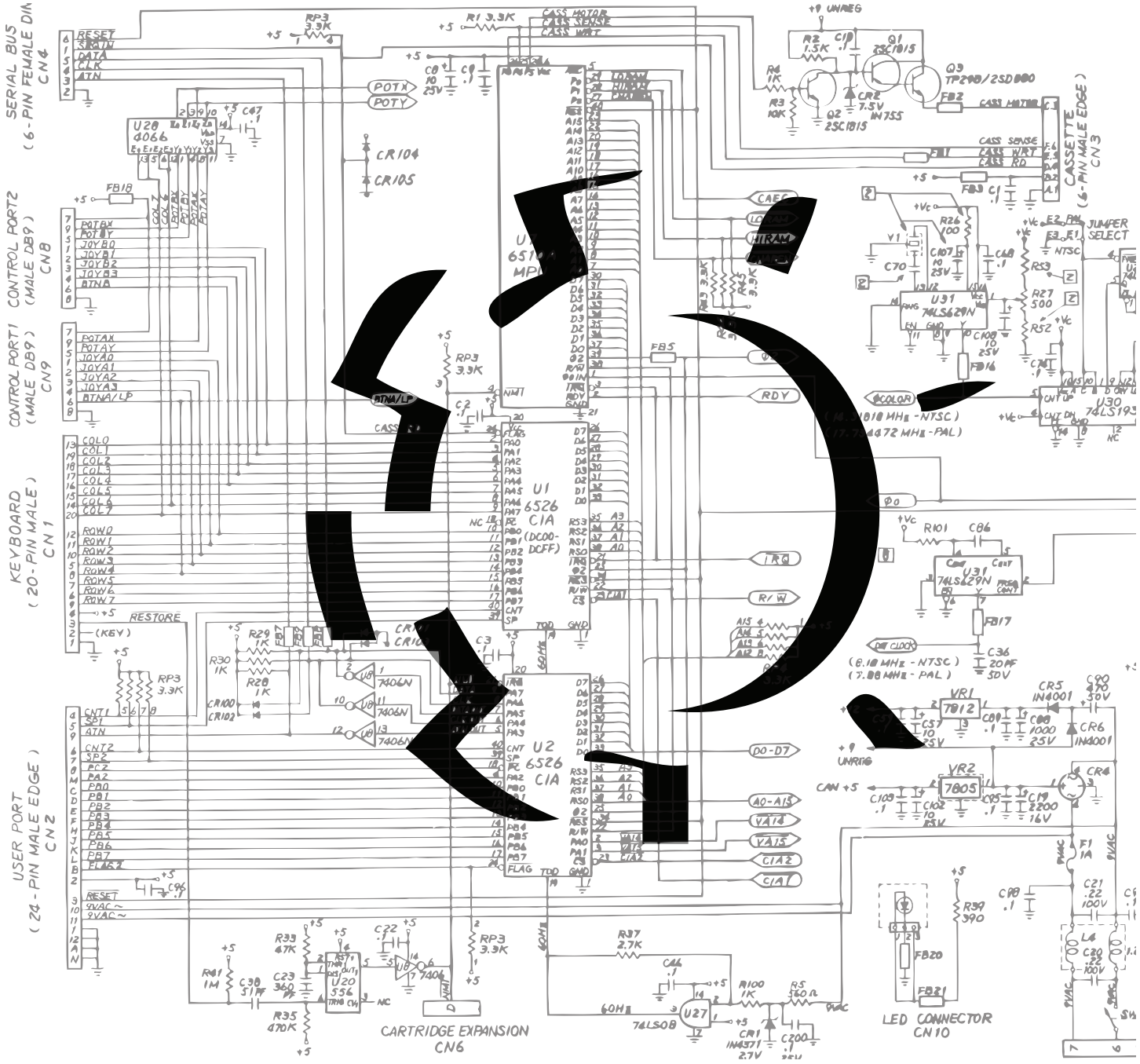

TABLE OF CONTENTS

Opening Thoughts

The idea for this project was to provide a means of publication for interesting articles. Not everyone likes to write tutorials, and not everyone feels that the information they have is enough to constitute a publication of any sort. We all run across interesting protections, new methods of debugger detection, and inventive coding techniques. We just wanted to provide the community with somewhere to distribute interesting, sometimes random, reversing information.

While the title of this ezine says ARTeam, we prefer to think that we are acting as a conduit. We really hope that you find this project interesting, and we really want this to be a community project. So if you have an idea for an article, or just something fascinating you want to share, let us know and hopefully we will see a ezine #2.
It soon became apparent that the scope of this project went well beyond what we had predicted. A big thanks goes out to all the contributors. Without you this would be a blank page. We also need to thank everyone who has viewed, refined and commented on the production of this ezine. Hopefully we have been able to provide the reversing community something interesting.

The reversing community has been very dynamic in the past few years. We've seen a ring3 GUI debugger grow in startling popularity. We've seen protection authors dig deeper into the OS in an effort to deter crackers. Unique protections have provided months of analysis for reversers. New inventive tools have been developed in the reversing community in an effort to effectively analyze and understand software protection. And ironically we see some of these tools move back to ring0.

None of these changes and achievements would have been possible without the amazing and talented reversers that take the time to share their knowledge and teach others. No matter what team you belong to, what level you reverse at, what language you speak, you all make up the same community. A group of people who constantly strive for discovery. None of us are content with accepting things "as they are" we need to know why. We are the scientists of software. We dig deeper than the average user, we see code where everyone else see flashy presentation. We learn this code so well that we can rewrite it, manipulate it, and even improve on it.

Since these are my thoughts, I just want to thank every single member of the reversing community. I couldn't even begin to name every single person who has provided a contribution. We are all spread out among many boards, many teams, even many countries. But I like to think that we all share a certain camaraderie.

Please enjoy the information included among these pages, we had some talented people give us some great submissions.

Gabri3l[ARTeam]

Interviewed: *Nilrem of ARTeam*

What first started your interest in Reverse Engineering?

Oh my! What a tricky question, there are numerous factors, however these other factors are actually the reasoning that kept my interested ignited but wasn't the initial fuel for the fire. If I'm been honest, I'd been using cracks/serial/keygens since I'd gotten the internet (1998), it was only when there was no crack out there for a certain program that I hit a brick wall. Do I wait a couple of days/weeks/months for a fairly obscure piece of software to be cracked? No of course not, I need it and I need it now, aha! I better go learn how to crack. That's what started my interest - my neediness.

How long have you been active in reverse engineering?

Since the question is how long I have been active in reverse engineering and not when did I initially start. The most accurate date I can give you for that question is when I wrote my first tutorial (obviously I would have been active before this because, of course, I had to learn how to crack before I could start tutorial writing). My first tutorial ever written was "Finding a hardcoded serial and patching the program to except any serial 01",and this was written on the 11th of August 2003. So take 11th of August 2003 as the answer the question.

What made you decide to form ARTeam?

A girl, a girl named Kyrstie, we had split up so I decided to start writing tutorials because of all the free time I now had.
When I first started writing tutorials I was publishing them on exetools. Which at the time was recieving little to no tutorial submissions as a result of this I started recieving a fair bit of attention. One of the people interested in me and what I was doing was PompeyFan (who subsequently became the Co-Founder heh). He sent me pms saying I had helped him on the road to Reverse Engineering and had asked me something along the lines of:
"Hi, Nilrem, your tutorials are great. When I am good enough can I join your team please?"
I'm guessing you can imagine my reaction, team...TEAM?! I don't have no team.. uhh, hang on a minute, brain-storm!!
That's how it happened, that is how ARTeam was born, someone liked my tutorials wanted to join my team so I started ARTeam so he could join, and the rest as they say, is history.

How did you end up with the original founders/members?

Well since my memory isn't the best, and I'm probably going to annoy a few staff members here by forgetting the order in which they joined. If I remember correctly the next addition to the family (no I'm not doing my Don

Corleone impression), was Ferrari. Who was actually reluctant to join because he didn't deem himself at an acceptable level of Reverse Engineering to join the team (damn what is it with these people heh).
So I had to wait for him to finish his 'training' from el-kiwi before he would join.
Now this is where it get's really hazy (Davy and Killer Joe?), the next few members to join where, MaDMAn_H3rCuL3s, Kruger, EJ12N, Enforcer, and Shub Nigurrath, these members became the initial core of ARTeam. Now how did they actually start with ARTeam? That is a very tricky question, so I'll avoid it. I do however know where I met them all (except Shub, we met on the ARTeam board through word of mouth), which is Exetools, so praise be to (Yevon?) Exetools.

What is your opinion on the ethical aspect of cracking / reversing?

Well I'll try not to write an essay alone on this question, not because I don't want to, but because there are numerous (to say the least) debates on this specific question.
You see you have put a slash between 'cracking' and 'reversing', whereas I see them as two different (similar but different) things. They differ because cracking to me implies everything that ARTeam is (no longer) not about, and 'reversing' is exactly what ARTeam is about (one facet of our ideoligies anyways). You see cracking (and label me hypocritical if you wish) is wrong and Reverse Engineering is right! That is if you see only in black and white which thankfully I don't (and even then RE would probably be deemed wrong, if so virii anaylzers please stop reverse engineering those virii).
First allow me to define cracking and Reverse Engineering.
Cracking (to me) just means releasing cracks (even by stealing other peoples work) to gain notoriety for oneself and ones group without giving (accept from the cracks) anything back to the community of which they learnt there appropriate skills.
Now Reverse Engineering entails the same process, we Reverse Engineer various softwares and their corresponding protection schemes and we then compile them into tutorials for people to learn. We actually give back to the community that gave us so much. Isn't this changing the question? No it is allowing me to start to answer (you like to ramble don't you? Yes, and coincidentally talk to myself) the question properly. Now you know my views on cracking and Reverse Engineering, you can now see (hopefully) why things aren't as black and white as the media, authorities, and software companies like to make out.
I personally do believe it is wrong to release cracks, then on the otherhand I don't believe it is wrong for a poor student to crack thousands of pounds worth of software so he can learn for free (Visual Studio for example). I certainly do not deem Reverse Engineering wrong, in fact what we are doing is helping people, and there is absolutely nothing wrong with that. We at ARTeam teach people to share their knowledge and to help others in a friendly and polite manner. What is wrong with that? Absolutely nothing! Once people understand that we are similar to anti-virus companies, in that we both Reverse Engineer to help people (our help isn't as obvious that's all), and that we aren't out to hurt anyone or their livelihood, then one day we might actually be praised by people outside of our communities (don't hold your breath though).

What do you find most interesting about the web scene right now?

If I understand your question correctly then you are referring to the cracking scene's websites.
What do I find most interesting, well I'll just pick one thing since it gives me an ego boost, and that is many different groups with forums are following suit with ARTeam. By this I mean they have turned into a tutorials only group. Actually that isn't an ego boost is it? No of course it isn't, we changed our policies for a different reason to the other groups I'm referring to. In fact it is quite saddening, they have changed their policies because their communities were starting to turn into war zones (exaggeration yes, but only because they changed their policies

just in time before things could escalate uncontrollably).
So you see it's interesting to see how the scene is changing, no longer is it "ahh thankyou for giving me that release", it is more like "You haven't cracked it within 35 seconds, you suck! I hate you!!", of course this is a obvious re-enactment because I used correct grammar. 8-)

Has anything you've learning during RE become useful in real life?

Yes and no. No not in any obvious ways, yes in obscure ways as a result of studying Reverse Engineering.
I have learnt how to program in assembly, which I never would have done without learning Reverse Engineering (because I needed it).
I have learnt how to communicate and express my ideas to others as a result of numerous discussions on ARTeam and tutorial writing.
I become more logic minded in the way I approach different problems which will no doubt help me with my games development studies.
I have met (virtually) lots and lots of talented people, but how does that help you Merlin?? Well if we meet in person one day hopefully they have a nice looking sister who will become my bride?
Ok ok so it's getting a bit far-fetched now, but as you can see it has helped me, just not in any blatant way until you start looking at it more in-depth.

What do you see the future of software protection being?

Longer sentences? Perhaps even the death penalty? I just really can't see how they will stop the 'crackers', even the death penalty wouldn't stop everybody. I believe they'll start using more hardware protection actually, but the question was software protection so I'll try to address that accordingly. Maybe they'll employ Reverse Engineers from certain teams (hint hint). All jokes aside, I believe software protection will get harder but that will only add more fuel to the fire of the Reverse Engineers out there. Basically I really have no idea on what the next step will be, but before Arma and Aspr no-one said. "Ahh yes this new protection will be [insert Arma and Aspr characteristics here]."
Hopefully that answers the question.

We've seen people all across the scene come and go, have you ever thought of "getting out"?

Yes you're right we have, some of those people were ARTeam members too, so the reality of people quitting or 'retiring' is very prominent. Have I ever thought of "getting out"? Yes, I have, and I did. It was last Summer, I was having personal issues and wanted to address them, and with a second life there I decided it would be easier to manage just one life.
As a result I did one of the hardest things I have ever had to do, not only say goodbye to the dream I started, but say goodbye to my new family, a very close-knit family at that as well.
But we never heard anything????!!! Ahh you see I did it quietly and privately with no public announcements. It also was a good thing my departue from ARTeam because it put to the test one of my theories. You see when ARTeam started I have always said that it was to be run as a true Democracy were every major change had to go through a majority vote wins scenario. So when I left the team carried on as normal and even went from strength to strength without me. Of course this made me sad and happy at the same time, my baby was no longer a baby and I wasn't needed, at the other end of the spectrum I had created something that could live and survive without

me. Not many other groups can make that claim when the founder leaves.
But you're here now? Yes I came back, I couldn't leave my family, not for long anyways. 8-)

Are there any comments you would like to add?

Yes, can't believe I've come to the end of the interview! Ha! It's been a pleasure it really has, I'm a lot more hungry then I was when I started the interview so I'm going to have to go eat. 8-P
I just want to say a big thankyou to everyone that has contributed to, and, helped in some way this very first issue of the Ezine. You have all worked incredibly hard (accept from me 8-P) and it shows.
Readers, thanks for, well, erm, reading. Look out for the next issue!

-Merlin

It's not hard to realize that systematic bruting is only realistic if you are bruting something with tremendous speed (server on your lan, or a hashed pw on your own computer).

So, our bruter will use dictionary bruting, it will take the path to the dictionary file as one of its command line parameters.

Next, we will want to write the code to build the "base packet."
A base packet is necessary for fast bruting - in our case the base packet should have the static data already in it - the only thing that should be left out is the crc and the password since those will change every time on a new attempt. Some bad bruters will make a new array every attempt which is slow and inefficient - allocating memory is time-consuming. Other bad bruters will have a "base packet" but rewrite the static content (command identifier, os, nick, etc) over and over again though it doesn't need to be.

If we are making a multi-threaded bruter, each thread should get its own base packet.

Here's the snippet of code from the src files used to make the base packet with comments about each line:

```
       packet = new byte[180]; Our packet size is 180 bytes
            MemoryStream stream = new MemoryStream(packet);
            BinaryWriter writer = new BinaryWriter(stream);
            //C# has no pointers - we use MemoryStream & BinaryWriter to write larger-than-
byte data to the packet
            writer.Write(new byte[]{
             0xF4, 0xBE, 0x03   We write the LOGIN command identifier
            });
            stream.Seek(80, SeekOrigin.Begin);  goto offset 80
            writer.Write((ulong)0x3C00020000002000);  write version
            stream.Seek(90, SeekOrigin.Begin);
            stream.WriteByte(0x02); write registered flag
            stream.WriteByte((byte)user.Length); write user length
            writer.Write(user.ToCharArray()); write user string bytes
            stream.Seek(150, SeekOrigin.Begin);  goto offset 150
            stream.WriteByte((byte)nick.Length); write nick length
            writer.Write(nick.ToCharArray()); write nick string bytes
```

In addition, when we were reversing the login packet we discovered that a string structure had 30 bytes - 1 for its length - 29 for its data.
This means any passwords from the password list with length greater than 29 should be dismissed.

The code for the TeamSpeak bruter I made in C# .NET (I used C# Express 2005 - it's free) is in the src folder that you should have received with this article
On some servers I get over 500 tries per second - UDP is fast! (http://en.wikipedia.org/wiki/User_Datagram_Protocol)

7. Conclusion:
Knowing how to reverse a protocol can be very useful whether you want to patch an online check or get the password of someone's X account. It can also provide an alternate way of cracking a prog: Instead of patching a program that implements an online check, you can write a loader that hooks onto the winsock api to modify the data the program receives from the server. This may result in a bad serial being accepted as a good serial.

You should now know:
- A protocol usually has an identifier for every type of action.
- The identifier is almost always the first few bytes of the packet.

- If the lower-level protocol used is UDP, the protocol most likely implements a checksum of sorts such as the CRC32.
- A secure protocol should have flood protection and SHOULD be encrypted by server-client key exchange.
- TeamSpeak's protocol is shit - reason being: we can write a bruter that is extremely fast and never gets banned for sending too many requests.

Be sure to checkout my AIM/AOL screenname bruter:
http://pop.pimpsofpain.com/showthread.php?t=5603&page=1&pp=10
and the included C# Project, UnTeamSpeak, a TeamSpeak bot that supports a variety of functions.

*Stay tuned for my next article in the ARTeam ezine which will feature an article on Reversing Gunbound's login protocol. Gunbound is a closed-source game that uses an encrypted protocol.

This article includes supplemental sources and files. They have been included with the ezine archive and can be found in the Supplements folder. Within the Supplements folder you will find a folder for each article that contains sources and files.

ARTeam eZine #2 Call for Papers

ARTeam members are asking for your article submissions on subjects related Reverse-Engineering.

We wanted to provide the community with somewhere to distribute interesting, sometimes random, reversing information. Not everyone likes to write tutorials, and not everyone feels that the information they have is enough to constitute a publication of any sort. I'm sure all of us have hit upon something interesting while coding/reversing and have wanted to share it but didn't know exactly how. Or if you have cracked some interesting protection but didn't feel like writing a whole step by step tutorial, you can share the basic steps and theory here. If you have an idea for an article, or just something fascinating you want to share, let us know.

Examples of articles are a new way to detect a debugger, or a new way to defeat a debugger detection. Or how to defeat an interesting crackme. The ezine is more about sharing knowledge, as opposed to teaching. So the articles can be more generic in nature. You don't have to walk a user through step by step. Instead you can share information from simple theory all the way to "sources included"

What we are looking for in an article submission:
1. Clear thought out article. We are asking you to take pride in what you submit.
2. It doesn't have to be very long. A few paragraphs is fine, but it needs to make sense.
3. Any format is fine.
4. If you include pictures please center them in the article. If possible please add a number and label below each image.
5. If you include code snippets inside a document other than .txt please use a monospace font to allow for better formatting
6. Anonymous articles are fine. But you must have written it. No plagiarism!
7. Any other questions you may have feel free to ask

We are accepting articles from anyone wanting to contribute. That means you. We want to make the ezine more of a community project than a team release. If your article is not used, its not because we don't like it. It may just need some work. We will work with you to help develop your article if it needs it.

Questions or Comments please visit http://forum.accessroot.com