Coremelt++

Maxfield Schuchard
Abedelaziz Mohaisen

In this project we aim to design and implement attacks against the data plane of the Internet by artificially creating instability in the BGP control plane. It has already been shown in [1] that changes that must propagate through the control plane of BGP cause increased packet loss and delays in the the data plane. In the past, these changes originate from rare events such as router misconfiguration or hardware failure, and in general are isolated in both location and time. We will use traffic generated by a botnet to artificially create these same events, unlike their naturally occurring counterparts, these events will be carefully clustered in both time and space. While the core of the Internet is tolerant of these events when they occur naturally, we predict that the core will not able to handle this "worst case scenario", in fact our attack uses the very mechanisms put in place to deal with localized faults to exacerbate the ill effects of our attack. The end result of this attack will be the same as the original Coremelt attack, however the resources required will be far lower.

While the end result of our attack will be similar in many ways to the standard distributed denail of service (DDoS) attack, it will different in a few key ways. Firstly the scope is larger, while standard DDoS aims to take down a specific set of targets, our attack is far less targeted, we aim to disrupt all traffic in the targeted portion of the Internet. While this attack is a cruder tool, it does have some advantages, for example the actual target of the attack will be hidden amongst the set of all victims. Secondly the destination of traffic for the attack is different, allowing this attack to bypass some currently released DDoS defenses such as [2] and potentially [3]. In this attack we will use the same traffic model as is outlined in the original Coremelt paper [4], in which, instead of sending large amounts of unwanted data to targets, bots will send large amounts of "wanted data" to other bots.

Motivation

The goal of this project is not to create a blueprint or how to guide on disrupting the Internet. The goal of this project is to investigate if there are potential vulnerabilities in the current state of BGP and propose solutions to them. The theoretical model of our attack revolves around a collection of BGP behaviors and policies, with a working simulation of this attack we would be better able to see which of these are cornerstones of the attack, and hopefully find ways to correct them within the scope of BGP. Looking at the longer term, we also wish to provide a set of guidelines and warning signs the developers of future inter-AS routing protocols will take into account in their design.

Related Work:

In the original Coremelt paper Studer and Perrig outlined an attack against the core of the Internet in which a botnet could saturate the bandwidth capacity of providers by simply sending data to other bots. The key fact their attack aims to exploit is the fact that providers oversubscribe, in other words they do not have enough bandwidth to handle all of their customers talking at once. The potential flaw in this attack is that the small providers that the bots are directly connected to are also oversubscribed, creating the possibility of saturating the fringes of the attack space, limiting the data bots could push into the core of the Internet. Their analysis revolved around a scale model of the Internet which was static, additionally the amount of over subscription is a closely guarded secret by most ISPs, forcing them to make best guesses at bandwidth capacities. These two facts raise issues about the validity of the original Coremelt attack. We avoid both of these issues in our attack. We will take dynamics of the Internet topology into account, in fact it is central to our attack. Additionally while bandwidth capacities are important to analyzing our attack, they are not a cornerstone, so potential inaccuracy is not as large of an issue.

There is a pre-existing body of work that focuses on events in the control plane of BGP creating loss of quality of service in the data plane [1, 5]. These papers serve as interesting snap shots showing what occurs in the data plane when a path changes. We will aim to springboard off of this research, using conclusion drawn from it to show which events provide the most "bang for our buck". Additionally there is a large body of work on BGP policy choices [6-10], which is central to our understanding and modeling of BGP behavior. Both of these bodies of research are critical to building a faithful simulation of Internet routing dynamics and to the design and implementation of Coremelt++.

The last area of related work that is interesting to us is that of current DDoS defenses. Papers such as [3, 4] build defenses that are used to protect services from DDoS, it will be interesting to look at how they perform against Coremelt++. In general we expect them to perform poorly since our attack aims to attack services indirectly by preventing users from accessing the Internet as a whole, instead of by attacking the service directly.

Methodology:

Obviously attempting to build a botnet and attempting actual attacks on the Internet is out of the question, no one involved here wants to end up in a cell in Leavenworth, Kansas. In order to research and measure the success of our attacks we will use simulations of the Internet, similar to the original Coremelt paper. Using data from CAIDA dataset we will construct a graph of ASes that is representative of the Internet. Literature on bandwidth capacities and traffic loads of ISPs will be used to create as accurate a picture of how these simulated ASes are connected. Again following in the methodology of Coremelt, we will scatter bots that we control throughout this graph, these will be used to create traffic that we use for our attack. From here we will branch off from the original Coremelt paper, we will give each of the ASes the ability to communicate with its neighbors using eBGP. Currently it is unclear if we will use an existing implementation of BGP or if we will create our own stripped down version of BGP. Policies in this simulated BGP environment will be driven by literature that lays out recommended best practices in BGP configuration, this literature was mentioned in our related work section.

Once the simulation environment is setup we can proceed to determine two things: which events on the control plane create the largest disruptions in the data plane, and which events on the control plane we can generate by our bot's actions on the data plane. We will be supported by some degree of existing literature on the first note. Some amount of additional investigation will need to be done still, since most literature looks at these events in isolation, and we wish to create several of these events, which might lead to different behavior.

In order to provide a measuring stick of the success of our attacks, we will measure the reductions in quality of service in traditional DDoS attacks with attackers of various resources. The main two metrics we will look at are packet loss and increased latency. After we have baselines for attackers of various resources, we will attempt to build attacks with the collection of control plane events we find we can produce above, and then compare the reduction in quality of service of attackers with equivalent resources. It is our assertion that we will be able to achieve the same quality of service degradation that a large botnet can achieve using naïve DDoS techniques with a far smaller botnet.

Management Plan:

Major Tasks:

1. Build test topologies (AS graph)
2. Implement dynamic routing between ASes (BGP)
3. Construct policies between ASes
4. Construct legitimate traffic/monitors (QoS degradation measures)
5. Collection naïve DDoS performance baselines
6. Construct model of control plane events that bots can induce
7. Construct model of data plane effects of control plane events
8. Design Coremelt++ DDoS per analysis of 6 & 7
9. Acquire Coremelt++ benchmarks and compare to the naïve DDoS

Time Table:

October 9th : Simulator topology construction functional and accurate.

October 23rd : Dynamic routing implemented and tested.

October 30th: Traffic patterns implemented, at this point only BGP policies of ASes remain.

November 4th : Simulator fully functional (topology/dynamic routing/AS policy/legitimate traffic) and tested, start the cataloging of data and control plane events and their effects, additionally start to measure naïve DDoS effects vs attacker resources measurements

November 6th : Wrap up naïve DDoS measurements

November 13th : Wrap up cataloging of data and control plane events, begin implementation of Coremelt++

November 27th : Final Coremelt++ implementation complete and functional.

December 2th : Data analysis completed, start major writing push.

December 9th : Paper finished

Bibliography:

[1] F. Wang, Z. M. Mao, J. Wang, L. Gao, and R. Bush, "A Measurement Study on the Impact of Routing Events on End-to-End Internet Path Performance," Proc. of ACM SIGCOMM'06, 2006.

[2] C. Dixon, T. Anderson, and A. Krishnamurthy, “Phalanx: Withstanding Multimillion-Node Botnets”, NSDI'08

[3] J. Chou, B. Lin, S. Sen, O. Spatscheck, "Proactive Surge Protection: A Defense Mechanism for Bandwidth-Based Attacks," IEEE/ACM Transaction on Networking

[4] A. Studer and A. Perrig, “The Coremelt Attack”, 2009.

[5] N. Feamster, H. Balakrishnan, J. Rexford, "Some Foundational Problems in Interdomain Routing", Proc. ACM SIGCOMM Workshop on Hot Topics in Networking (HotNets-III). San Diego, CA, November 2004.

[6] L. Gao and J. Rexford, “Stable Internet Routing without Global Coordination,” ACM SIGMETRICS 2000, also IEEE/ACM Trans. Networking, vol. 9, pp. 681–692, December 2001.
[7] T. G. Griffin, F. B. Shepherd and G. Wilfong, "The Stable Paths Problem and Interdomain Routing", IEEE Transactions on Networking, April 2002.
[8] N. Feamster, J. Borkenhagen, and J. Rexford, "Guidelines for interdomain traffic engineering," ACM SIGCOMM Computer Communications Review, October 2003.
[9] S. Lee, Y. Yu, S. Nelakuditi, Z. Zhang, C. Chuan, "Proactive vs Reactive Approaches to Failure Resilient Routing", IEEE INFOCOM 2004.
[10] M. Caesar and J. Rexford, "BGP Routing Policies in ISP Networks", IEEE Network Magazine, special issue on interdomain routing, November/December 2005.