Maschinen hacken Maschinen

Ein Blick in die Zukunft der digitalen Angriffe

Bei der DARPA Cyber Grand Challenge traten sieben Supercomputer gegeneinander an und hackten sich gegenseitig. Künftig sollen so durch Maschinenunterstützung Bugs schneller aufgespürt werden können. Was aber, wenn die schiere Kraft des Computers zum massenhaften Produzieren von Exploits verwendet wird?

Von Uli Ries

Mancher sieht sich schon am Vorabend der Machtübernahme durch Skynet, der wild gewordenen KI der Terminator-Filme: Software untersucht Software auf Fehler, Computer hacken einander, Software testet Updates, zieht sie im Zweifel zurück und installiert sie letztendlich. Willkommen bei der DARPA Cyber Grand Challenge (CGC), dem ersten Hacker-Wettbewerb ohne menschliche Hacker.

Insgesamt 55 Millionen US-Dollar investierte die DARPA (Defense Advanced Research Projects Agency), der zivile Forschungsarm des US-Militärs, in den Wettbewerb. Letztlich will man mehr Sicherheit in die sich ständig stärker vernetzende Welt bringen, die voll von fehlerhafter Software ist und die bislang ausschließlich manuell gegen Attacken verteidigt wird. Daher haben Angreifer gut ein Jahr, bis eine von ihnen bereits missbrauchte Lücke entdeckt wird und ein Patch bereitsteht, erläutert Mike Walker, Programm-Verantwortlicher für die CGC. Diese lange Zeitspanne sollen autonom arbeitende Systeme verkürzen. Idealerweise sinkt die Reaktionszeit so auf Minuten oder gar Sekunden.

Autonom agierende Software ist ein komplexes Unterfangen, deshalb wählte die DARPA das Format der Grand Challenge, also eines Forschungsvorhabens, das mehrere Jahre dauert und internationale Beteiligung verlangt. Die CGC fährt quasi im Windschatten der Grand Challenges beziehungsweise Urban Challenge für autonomes Fahren aus den Jahren 2004, 2005 und 2007. Sie gelten als Keimzelle der momentan so heiß disku-

© Copyright by Heise Medien
Persönliches PDF für alle Leser der c't aus 30625 Hannover

tieren Autopiloten in Autos. Die CGC-Systeme sollen einen ersten Beleg fürs Funktionieren autonom hackender Rechner schaffen.

Der Wettstreit beginnt

In zwei Runden qualifizierten sich die sieben Programme für das Finale, das gut zehn Stunden lang dauerte und am Vortag der Hacker-Konferenz Def Con in Las Vegas stattfand. Die sieben Teams hatten ein Jahr lang Zeit, ihr sogenanntes Cyber Reasoning System (CRS) zu programmieren. Die DARPA stellte ihnen dafür und für das eigentliche Finale einen Supercomputer zur Verfügung, der aus 64 Nodes mit je zwölf Dual-Core-Xeon-Prozessoren (2,5 GHz) bestand und 16 Terabyte RAM hatte. Für das Finale wählten die Verantwortlichen das von der Def Con und anderen Wettbewerben bekannte Format des Capture the Flag (CTF): Alle Teams bekommen die gleiche, mit Bugs präparierte und damit verwundbare Software, schreiben Patches zum Selbstschutz und starten Angriffe gegen die anderen Mannschaften. Für jede erfolgreiche Aktion und für jeden erreichbaren Dienst gibt es Punkte. Der große Unterschied der Grand Challenge zu allen bisherigen CTF-Wettstreits: Die Programmierer sitzen an der Seitenlinie und schauen ihrer Maschine zu, wie sie sich gegen die anderen schlägt.

Für das Finale der CGC mietete die DARPA einen riesigen Ballsaal im Def-Con-Konferenzhotel und stellte die sieben Maschinen – samt sieben weiterer Supercomputer, die den menschlichen Schiedsrichtern zur Bewertung der Ergebnisse dienten – auf eine Bühne. Die Teams saßen davor und sahen live durch aufwendige Visualisierungen, was in jeder der 96 Spielrunden vor sich ging. In Art zweier Sportmoderatoren diskutierten ein Wissenschaftler und ein DARPA-Vertreter die jeweiligen Rundenergebnisse. In jeder Runde galt es eine andere Aufgabe zu lösen, ganz in CTF-Tradition.

Die zu untersuchenden Binärdateien liefen innerhalb des von der DARPA eigens für den Wettbewerb geschriebenen Betriebssystems namens DECREE (DARPA Experimental Cy-

Mike Walker zeichnet für den CGC-Wettbewerb verantwortlich.

Anzeige

© Copyright by Heise Medien
Persönliches PDF für allen Leser der c't aus 30625 Hannover

Im Unterschied zu menschlichen Hackern, denen ab und zu ein Softdrink zur Abkühlung genügt, müssen die Maschinen mit schwerem Gerät gekühlt werden.

bersecurity Research Evaluation Environment). Das System gibt es als virtuelle Maschine unter repo.cybergrandchallenge.com zum Download. Um die Analyse der Wettbewerbsergebnisse zu vereinfachen, gehören zu DECREE lediglich sieben Systemaufrufe. Die Binärdateien sind zwar x86-kompatibel, bringen aber ihr eigenes Binärformat mit und haben keine Gemeinsamkeiten mit in der übrigen Softwarewelt verwendetem Code oder Protokollen. Außerdem sorgt DECREE für einen hohen Grad an Reproduzierbarkeit, so dass sich während des Wettbewerbs erzeugte Codeteile leicht nachvollziehen lassen. Man wollte die Hackersoftware zudem nicht auf ein gängiges Betriebssystem loslassen, um zu verhindern, „dass die Bots binnen weniger Stunden sechshundert Bugs finden, die dem betroffenen Hersteller dann viel Kopfschmerzen bereiten“, sagte Walker.

Die durch Decree gebotene Reproduzierbarkeit ist wichtig für das Bewerten der von den Maschinen erzeugten Angriffe: Die CRS müssen keine fertigen Exploits für entdeckte Schwachstellen schreiben, sondern den Schiedsrichtern schlüssig beweisen, dass ein bestimmter Codeteil für einen Angriff nach einem ebenfalls zu beschreibenden Muster anfällig ist.

And the winner is …

Nach gut zehn Stunden setzte sich das Team namens ForAllSecure von der Carnegie Mellon University gegen die Konkurrenz durch und nahm für seine Mayhem getaufte Software zwei Millionen Dollar Preisgeld mit nach Hause. Die Zweitplatzierten vom Team TechX immerhin noch eine Million, Team Shellphish mit seiner Mechanical Phish getauften Software 750.000 US-Dollar. Shellphish ging nach Auskunft des Team-Leiters Professor Giovanni Vigna das ganze Projekt ziemlich chaotisch an: Man bewarb sich wenige Sekunden vor Ende der Frist überhaupt erst für die Teilnahme. Die zwölfmonatige Vorbereitung ließen die angehenden Wissenschaftler und ihr Professor beinahe komplett ungenutzt verstreichen und arbeiteten dann die letzten drei Wochen quasi Tag und Nacht an Mechanical Phish. Vigna sagte lachend: „Was soll auch anderes dabei herauskommen, wenn man einen Haufen Master-Studenten unter der Leitung eines Italieners arbeiten lässt.“

»Die gefundenen Bugs sind meine Altersversorgung.«

Teamleiter David Brumley

Shellphish veröffentlichte sein zur Binäranalyse verwendetes Framework namens Angr noch während der Def Con auf Github. Angr setzt auf einen Mix aus statischer und symbolischer dynamischer Ausführung und wies während des Finales beispielsweise einen funktionierenden Angriff auf eine Abwandlung des berühmt gewordenen Sendmail-Bugs Crackaddr nach, den die Organisatoren in den Binärdateien versteckt hatten.

Letztendlich hielten alle sieben Maschinen bis zum Schluss durch. Insgesamt lieferten sie 650 Belege für Schwachstellen und reichten 421 Patches ein – in Software, die sie nie zuvor analysieren konnten. Wenngleich dennoch Unvorhergesehenes passierte: Die Mayhem-Software von ForAllSecure reagierte über mehrere Dutzend Runden nicht auf die sogenannten Polls, mit denen die Schiedsrichter Verfügbarkeit und Performance der einzelnen Dienste auf den Maschinen prüften. Die Nichtverfügbarkeit ging laut David Brumley von ForAllSecure auf einen simplen Grund zurück: Die Programmierer hatten die Netzwerklast unterschätzt, die Gegner und Schiedsrichter während des Wettbewerbs erzeugten. Mayhem ertrank quasi in Paketen.

Warum Mayhem dennoch gewann, erklärte David Brumley im Gespräch mit c’t so: „Wir haben uns beim Programmieren darauf konzentriert, Mayhem möglichst schlanke Patches schreiben zu lassen.“ Das war wichtig im Wettbewerb, da die Schiedsrichter nicht nur die Verfügbarkeit sämtlicher Dienste auf den sieben Teilnehmersystemen prüften, sondern auch deren Performance. Ein allzu ungelenker Patch hätte daher für Punktabzug in allen folgenden Runden gesorgt. Außerdem zählte die Zeit, die die zu patchende Funktion während des Updates offline war. Das Team löste also die schwierigste der Aufgaben am besten und gewann.

Im Wettstreit gegen menschliche Hacker ging Mayhem dann dennoch unter: Das Team schickte die Maschine in den CTF-Wettbewerb der Def Con, wo sie gegen 13 menschliche Gruppen und ein hybrides Team aus Shellphish und Mechanical Phish antrat – und letzter wurde. Verantwortlich für die Niederlagen war nicht nur, dass Menschen erheblich raffinierter – und teilweise unfair – ans Werk gehen und sich Fallen stellen während des Wettbewerbs. Zudem mussten die Programmierer der CRS ihre Software während des Wettstreits an das API des menschlichen CTFs anpassen und verloren somit wertvolle Zeit.

Was Mayhem in der wirklichen Software-Welt zu leisten vermag, verriet David Brumley im Gespräch: Man habe das CRS im Vorfeld zu Testzwecken innerhalb von 24 Stunden 35.000 Linux-Binärdateien untersuchen lassen. Die Software fand mehrere Tausend Bugs, die es nun auf ihren Schweregrad hin zu untersuchen gilt. Bug-Jäger aus Fleisch und Blut können angesichts dieser Zahl nur vor Neid erblassen. Was Brumley mit den Bugs vorhabe? „Sie sind meine Altersversorgung“, sagte er lachend. Wenngleich das nur halb als Scherz zu verstehen ist: ForAllSecure veröffentlicht den Quelltext seines CRS nicht und plant, die Software in Zukunft kommerziell einzusetzen. Dies dürfte das Bug-Hunting ziemlich schnell auf den Kopf stellen, wenn die Maschine im Zusammenspiel mit menschlichen Fachleuten auf Fehlerjagd geht. *(fab@ct.de)* c't

© Copyright by Heise Medien
Persönliches PDF für alle Leser der c't aus 30625 Hannover