ივ. ჯავახიშვილის სახ. თბილისის სახელმწიფო უნივერსიტეტი
ზუსტ და საბუნებისმეტყველო მეცნიერებათა ფაკულტეტი

გიორგი ადამოვი

ქსელის უსაფრთხოების უზრუნველყოფის შეფასება

სამაგისტრო პროგრამა ინფორმაციული ტექნოლოგიები

ნაშრომი შესრულებულია ინფორმაციული ტექნოლოგიების
მაგისტრის აკადემიური ხარისხის მოსაპოვებლად

ხელმძღვანელი: ზურაბ მოდებაძე

ფიზ. - მათ. მეცნ. კანდ.

ქ. თბილისი

2014 წ.

სარჩევი

ანოტაცია

ნაშრომის მთავარი მიზანია ქსელის უსაფრთხოების შეფასების დროს გამოყენებული მეთოდებისა და ინსტრუმენტების აღწერა-ანალიზი. ძირითად ნაწილში შეთავაზებულია ქსელის უსაფრთხოების შეფასების პროცესის გაუმჯობესების მეთოდები და განხილულია ინსტრუმენტები, რომელიც ყველაზე ხშირად გამოიყენება ქსელის არასანქცირებული შეტევების დროს. ეს განსაკუთრებით მნიშნელოვანია კორპორაციებზე ქსელური შეტევების ზრდის ფონზე და ხალხის შეშფოთებას იწვევს პირადი ინფორმაციის დაუცველობა. შესრულებული ნაშრომის მთავარი შედეგია აპლიკაციის პროტოტიპი Android სისტემისთვის, რომელიც დაფუძნებულია ქსელის ერთ-ერთ ყველაზე პოპულარული სკანირების ინსტრუმენტზე, ეს აპლიკაცია დაეხმარება ქსელის უსაფრთხოების სპეციალისტებს სამიზნე ქსელის სწრაფ ანალიზში.

The main objectives of this work are to analyze and describe the methodologies and tools of network security assessment process. Most work is concentrated on suggesting a means for network security assessment process improvement and review of tools most usually used in hacking attempts. This is especially important in the light of growing network attacks on corporations and public concern about private information security. Main result of this thesis is an application prototype for Android systems, based on one of the most popular security scanning tool, which is expected to help security assessment specialist to quickly analyze the targeted net host.

შესავალი

ინტერნეტი არის გლობალური კომპიუტერული ქსელი, რომელიც ეფუძნება IP პროტოკოლსა და პაკეტთა მარშრუტიზაციას. იგი ქმნის გლობალურ საინფორმაციო სივრცეს და წარმოადგენს მსოფლიო ქსელის საფუძველს.

Internetworldstats.com-ის მიერ გამოქვეყნებული სტატისიკის მიხედვით საქართველოში ინტერნეტის მომხმარებელთა რიცხვი ბოლო წლებში საგრძნობლად გაიზარდა. მათ აქვთ შედარებული 2000 წლისა და 2010 წლის მონაცემები, რომელთა შედეგების ნახვა შესაძლებელია სურათზე:

YEAR	Users	Population	% Pen.
2000	20,000	4,389,004	0.5 %
2006	332,000	4,389,004	7.6 %
2009	1,024,000	4,615,807	22.2 %
2010	1,300,000	4,600,825	28.3 %

მომხმარებელთა რაოდენობისა და ინტერნეტ სერვისების გაზრდასთან ერთად იზრდება ქსელის გატეხვის რისკები, რადგან უფრო და უფრო მეტი ინფორმაცია გროვდება სერვერებსა და მომხმარებელთა კომპიუტერებში, რომლებიც ჩართულნი არიან ქსელში. ინტერნეტ უსაფრთხოების სფეროში ბოლო 5 წლის განმავლობაში მომხდარმა სკანდალებმა ნათლად აჩვენა ქსელის უსაფრთხოების შეფასების, დაცვისა და ასევე ამისთვის ახალი ინსტრუმენტების შექმნის აუცილებლობა.

ბოლო წლის განმავლობაში ჩვენ ასევე ვხედავთ ინტერნეტ ტექნოლოგიების აქტიურ განვითარებას მობილური ტექნოლოგიების მიმართულებით. კომპიუტერული ტექნიკის უფრო და უფრო მეტი მწარმოებელი გადადის ჩვეულებრივი desktop ტექნოლოგიებიდან მობილურ მოწყობილობებზე. ასევე სწრაფი ტემპით იცვლება ინტერნეტის როლი ადამიანების ყოველდღიური ცხოვრებაში. თუ ადრე ინტერნეტი იყო რაღაც უცნაური და მიუწვდომელი, ახლა ყველას შეუძლია მისი გამოყენება და ასე თუ ისე ყოველღიურად იყენებს მას. ადამიანების უმრავლესობას ყოველთვის აქვს ჯიბეში მობილური ტელეფონი ან პლანშეტი, რომლის მეშვეობით შესაძლებელია წამებში ნებისმიერი ინფორმაციის მოპოვება გლობალური ინტერნეტ ქსელიდან. მობილური ტექნოლოგიები გახდა ჩვენი ცხოვრების ნაწილი.

ტექნოლოგიების განვითარებასთან ერთად თვითონ ქსელები, რომლებიც უზრუნველყოფენ ინფორმაციის მიწოდებას, გახდნენ ბევრად უფრო რთული და ვრცელი ვიდრე ადრე და როგორც წესი, უფრო დაუცველნი სხვადასხვა სახის თავდასხმებისაგან.

დღეს ნებისმიერ ადამიანს, რომელმაც იცის ქსელის მუშაობის პრინციპები, შეუძლია ჩამოტვირთოს შესაბამისი ინსტრუმენტები ღია ქსელიდან, ვთქვათ კაფეში, და მოიპაროს

ამავე ქსელში შეერთებული აბონენტების პირადი ინფორმაცია. ასევე ბოლო წლების განმავლობაში გაიზარდა DDoS თავდასხმების რაოდენობა დიდ კომპანიებზე და ამასთან ერთად კომპანიების მონაცემების მოპარვის შემთხვევები.

მობილური ქსელების სწრაფი განვითარების პირობებში ქსელის უსაფრთხოების სპეციალისტებს სჭირდება უფრო ეფექტური და მობილური ინსტრუმენტები ქსელის უსაფრთხოების სწრაფი ანალიზისათვის და საჭიროების შემთხვევაში სატესტო შეტევების განხორციელება იმ ინსტრუმენტებით, რომლებსაც იყენებენ ჰაკერები.

მობილური მოწყობილობებისთვის კერძოდ, Android-ის სისტემის მქონე სმარტფონებისათვის, არსებობს სხვადასხვა უსაფრთხოების ინსტრუმენტები, რომლებიც ეხმარებიან ქსელის ანალიზში და აჩვენებენ ქსელის უსაფრთხოების ხარვეზებს. მაგრამ ამ ინსტრუმენტების უმრავლესობა არის console აპლიკაციები, რომლებიც ითხოვენ არამარტო თვითონ ინსტრუმენტის, არამედ ასევე სისტემის command line-ის გამოყენების ცოდნას.

უსაფრთხოების სპეციალისტების მუშაობის ხელშეწყობის მიზნით აუცილებელია არა მხოლოდ უსაფრთხოების შეფასების ახალი ინსტრუმენტების, არამედ ამ ინსტრუმენტებისათვის მოსახერხებელი ინტერფეისების შექმნა.

ქსელის უსაფრთხოების საჭიროება

ადრე ჰაკერები იყვნენ მაღალკვალიფიციური პროგრამისტები, რომლებიც ძალიან კარგად ერკვეოდნენ კომპიუტერული კომუნიკაციების დეტალებში და იმაში, თუ როგორ შეილება გამოიყენო ქსელის „წყვილადობები“. დღეს ნებისმიერ ადამიანს შეუძლია ჩამოტვირთოს შესაბამისი გატეხვის პროგრამა და გახდეს ჰაკერი. ამ რთულმა შეტევის (თავდასხმის) პროგრამებმა, ინსტრუმენტებმა და ზოგადად ღია ქსელებმა, გამოიწვია ქსელის უსაფრთხოების საჭიროების გაზრდა და დინამიური უსაფრთხოების პოლიტიკის ახალი მოდელების შექმნა. ყველაზე მარტივი გზა იმისათვის, რომ დავიცვათ ქსელი გარე თავდასხმებისაგან ესაა საერთოდ დავხუროთ ეს ქსელი გარე სამყაროდან. დახურული ქსელი აძლევს კავშირის უფლებას მხოლოდ სანდო ცნობილ მოწყობილობებთან და საიტებთან და არ აძლევს უფლებას გარე ქსელებთან დააკავშიროს ის. რადგან დახურულ ქსელებს არ აქვთ კავშირი ინტერნეტთან, ისინი უფრო დაცულნი არიან გარე შემოტევებისაგან. თუმცა მაინც არსებობენ შიდა საფრთხეები. ქსელების დაახლოებით 60-80%-ის ბოროტად გამოყენება ხდება კორპორაციის შიგნიდან.

ბოლო 20 წლის განმავლობაში დიდი, ღია ქსელების განვთარებასთან ერთად გაიზარდა საფრთხეების რაოდენობა. ჰაკერებმა აღმოაჩინეს უფრო მეტი ქსელის მოწყვლადობები და რადგან ახლა შესაძლებელია ისეთი ჰაკერული პროგრამების ჩამოტვირთვა, რომელთაც არ სჭირდება დიდი ტექნიკური ცოდნა და აპლიკაციები, რომლებიც იყენებდნენ ქსელს troubleshooting-სა და ოპტიმიზაციისთვის, შეიძლება გამოყენებული იყოს შეტევისთვის და შეიქმნას საფრთხეები.

უსაფრთხოების რისკის შეფასების აუცილებლობის მიზეზები

ორგანიზაციებს აქვთ ბევრი მიზეზი ინფორმაციის უსაფრთხოების საკითხების აქტიური და განმეორებითი განხილვებისთვის. პერსონალური მონაცემებისა და ასევე ზოგადად საზოგადოების უსაფრთხოების მოთხოვნების დაცვის მიზნით ყველა ზომის კომპანია ვალდებულია გამოიჩინოს მაქსიმალური ყურადღება და პრიორიტეტი ინფორმაციის უსაფრთხოების რისკების განხილვაზე.

IT უსაფრთხოების რისკის შეფასებას შეიძლება ჰქონდეს ბევრი სახელი, განსხვავებული ტერმინები, მოცულობა და მეთოდები, მაგრამ დაცვა ხდება ერთი მიზეზის გამო: ორგანიზაციის ინფორმაციული აქტივების რისკების განსაზღვრა და იდენტიფიცირება. ეს ინფორმაცია გამოიყენება რისკების შემცირების საუკეთესო მეთოდის განსაზღვრისა და ორგანიზაციის მისიის ეფექტური მხარდაჭერისათვის.

უსაფრთხოების რისკების შეფასების ზოგიერთი მიზეზია:

დანერგვის ღირებულება - უსაფრთხოების გაძლიერება ხშირად იწვევს დამატებით ხარჯებს, რადგან უსაფრთხოების გაზრდა ზრდის განსაზღვრულ შემოსავალს, ხარჯების გამართლება ხშირად რთულია. IT უსაფრთხოების ეფექტური შეფასების პროცესში მთავარ ბიზნეს-მენეჯერს უნდა აცნობონ ტექნოლოგიის გამოყენების ყველა კრიტიკული რისკის შესახებ და უშუალოდ შეთავაზონ უსაფრთხოების ინვესტიციების გამართლება.

პროდუქტიულობა - საწარმოს უსაფრთხოების რისკების შეფასებებით უნდა გაუმჯობედეს IT ოპერაციების, უსაფრთხოებისა და აუდიტის პროდუქტიულობა. რისკების შეფასებების ფორმირებული მიმოხილვა, მიმოხილვის სტრუქტურიზაცია, უსაფრთხოების ცოდნის ბაზის შექმნა და თვითანალიზის განხორციელება კომპანიის პროდუქტიულობის გაზრდის საშუალებას იძლევა.

ბარიერების მოხსნა - იმისათვის, რომ შედეგი მაქსიმალურად ეფექთური იყოს ორგანიზაციის მენეჯმენტი და IT პერსონალი ერთად უნდა მუშაობდნენ უსაფრთხოების საკითხებზე. ორგანიზაციის მენეჯმენტი პასუხისმგებელია გადაწყვეტილების მიღებაზე ორგანიზაციის სათანადო დონის უსაფრთხოების შესახებ. IT პერსონალი შესაბამისად პასუხისმგებელია სისტემებისათვის კონკრეტული უსაფრთხოების მოთხოვნების, პროგრამების, მონაცემების და კონტროლის შესახებ გადაწყვეტილების მიღებაზე.

თვითმმართველობის ანალიზი - საწარმოს უსაფრთხოების რისკის შეფასების სისტემა უნდა იყოს გასაგებად საკმარისად მარტივი, უსაფრთხოების და IT წოდნის გარეშე ადვილი გამოყენებისთვის. ეს შესაძლებლობას აძლევს მენეჯმენტს თავის ხელში აიღოს ორგანიზანიზაციის სისტემების უსაფრთხოების, პროგრამებისა და მონაცემების მართვა. ეს ასევე იძლევა საშუალებას, რომ უსაფრთხოება გახდეს უფრო მნიშნელოვანი ნაწილი ორგანიზაციის კულტურაში.

კომუნიკაცია - ორგანიზაციის სხვადასხვა ნაწილებიდან ინფორმაციის მიღება საკომუნიკაციო გადაწყვეტილების მიღების პროცესში უსაფრთხოების რისკის შეფასებას ხელს უწყობს.

შეტევების ტიპები

ქსელზე თავდასხმა ან უსაფრთხოების ინციდენტი არის ისეთი საფრთხე, შეჭრა, მომსახურებაზე უარის თქმა ან ქსელის სხვათავდასხმა ინფრასტრუქტურაზე, რომელიც აანლიზებს ქსელს და იღებს ინფორმაციას, რომელიც შემდგომში შეილება გამოყენებულ იქნას ქსელის დაზიანებისთვის. ბევრ შემთხვევაში ჰაკერი შეიძლება იყოს დაინტერესებული არა მარტო პროგრამების ბოროტად გამოყენებაში, არამედ ქსელური მოწყობილობების არასანქცირებული წვდომის მიღებაში. ორგანიზაციებში ინფორმაციის გაჟონვის ძირითად წყაროს, როგორც წესი, წარმოადგენენ შეუმოწმებელი ქსელური მოწყობილობები. ბევრ ორგანიზაციაში ყველა email-ი, ყველა ვებ გვერდის მოთხოვნა, ყველა მომხმარებელის სისტემაში შესვლა და ყველა გამოგზავნილი ფაილი მუშავდება რაღაც ქსელური მოწყობილობით. ასევე არსებობს ისეთი რეგულაციები რომლების ქვეშ სატელეფონო სერვისები და ხმოვანი შეტყობინებების სისტემები ასევე ჩართულნი არიან

ქსელში. თუ ჰაკერი „ფლობს“ ქსელის მოწყობილობებს ის „ფლობს“ მთლიან ამ მოწყობილობების ქსელს.

ქსელური თავდასხმების გავრცელებისა და პროგრამული უზრუნველყოფის პლატფორმის ტიპის თავდასხმის კლასები შეიძლება შეიცავდეს პასიურ მონიტორინგს, აქტიურ ქსელურ თავდასხმებს, close-in თავდასხმებს, ინსაიდერების მიერ ექსპლუატაციასა და მომსახურების მიმწოდებლის მეშვეობით თავდასხმებს. საინფორმაციო სისტემები და ქსელები არიან მიმზიდველი თავდასხმის სამიზნე და უნდა იყონ მდგრადი შეტევის აგენტების, ჰაკერებია და ქვეყნების მხრიდან წამოსული ყველა საფრთხის მიმართ. სისტემას უნდა ჰქონდეს საშუალება შეზღუდოს დაზიანებები შედეგები და სწრაფად აღვადგინოს მუშაობა შესრულებული შეტევის შემდეგ.

ძირითადად შეტევები იყოფა ორ კატეგორიად: „პასიური“- როდესაც ჰაკერი უსმენს ქსელში მოგზაური ინფორმაცია და „აქტიური“ - როდესაც ჰაკერი წყვეტს ქსელის ნომალურ ფუნქციონირებას.

პასიური შეტევა (passive attack) - პასიური შეტევის დროს ხდება ქსელის დაუცველი ტრაფიკის მონიტორინგი იმისათვის, რომ იპოვონ და ამოიღონ პაროლები და ინფორმაცია, რომელიც შეიძლება გამოყენებული იყოს სხვა ტიპის შეტევების განახორციელებლად. პასიური შეტევები მოიცავენ ტრაფიკის ანალიზს, არადაცული კომუნიკაციების მონიტორინგს, სუსტად დაშიფრული ტრაფიკის დეშიფრაციასა და ისეთი აუტენთიფიკაციის ინფორმაციის მიღებას, როგორიცაა მაგალითად პაროლები ან ელექტრონული მისამართები. ქსელის ოპერაციების პასიური შეტევა მოწინააღმდეგეებს საშუალებას აძლევს დაინახონ მომავალი ქმედებები. პასიური შეტევა იწვევს ინფორმაციის ან მონაცემთა ფაილების გამჟღავნებას მომხმარებლის თანხმობის ან ცოდნის გარეშე.

აქტიური შეტევა (active attack) - აქტიური შეტევის დროს ჰაკერი ცდილობს დაცულ სისტემეზე შეტევას. ეს შეიძლება გაკეთდეს სტელსით, ვირუსებით, ჭიებით ან ტროას ცხენების გამოყენებით. ეს შეტევები მიმართულნი არიან ქსელის ხერხემალზე (backbone) და ცდილობენ სატრანზიტო ინფორმაციის გამოყენებას ან მოახდინონ დისტანციური თავდასხმა ავტორიზებულ მომხმარებელზე. აქტიური შეტევა იწვევს მონაცემთა ფაილების გამჟღავნებას ან გავრცელებას, DoS-ის მატებას ან მონაცემების მოდიფიკაციის ცვლილებას.

განაწილებული შეტევა (distributed attack) - განაწილებული შეტევის დროს ჰაკერმა უნდა შეთავაზოს სისტემას რაიმე კოდი, მაგალითად Trojan hourse-ს ან back-door პროგრამა, ისეთნაირად, რომ სისტემამ ჩაითვალოს ეს კოდი დაცულად და შემდგომ გაანაწილოს ის სხვა კომპანიებსა და მომხმარებლებს შორის. ასევე განაწილებული შეტევები ფოკუსირებენ ტექნიკის ან სოფტის ზიანისმომტანი მოდიფიკაციას ქარხანაში ან დისტრიბიუციის დროს. ზიანისმომტანი კოდი შემდგომში გამოიყენება იმისათვის, რომ მიიღოთ არაავტორიზებული წვდომა ინფორმაციასთან ან სისტემური ფუნქციებთან.

Insider-ით შეტევა - insider-თა შეტევის დროს კომპანიის თანამშრომელი ახორციელებს შეტევას სისტემაზე. insider შეიძლება იყოს ზიანის მომტანი ან არა. ზიანისმომტანი insider შეგნებულად ისმენს, იპარავს ან აზიანებს ინფორმაციას, იყენებს მას არაკანონიერად ან ბლოკავს სხვა ავტორიზებულ მომხმარებლებს. არაზიანისმომტანი შეტევები ხშირად ხდება დაუდევრობის, არცოდნისა ან უსაფრთხოების გაუთვალისწინებლობის გამო დავალების შესრულების დროს.

Close-in Attack - Close-in შეტევის დროს ჰაკერი ცდილობს ფიზიკურად მიუახლოვდეს ქსელის კომპონენტებს, მონაცემებსა და სისტემებს რათა მეტი გაიგოს ქსელის შესახებ. ამ შეტევის დროს იგი უნდა მიუახლოვდეს სისტემებსა და data-ცენტრებს იმისათვის, რომ შეცვალოს, შეკრიბოს ან დაბლოკოს ინფორმაციაზე წვდომა.

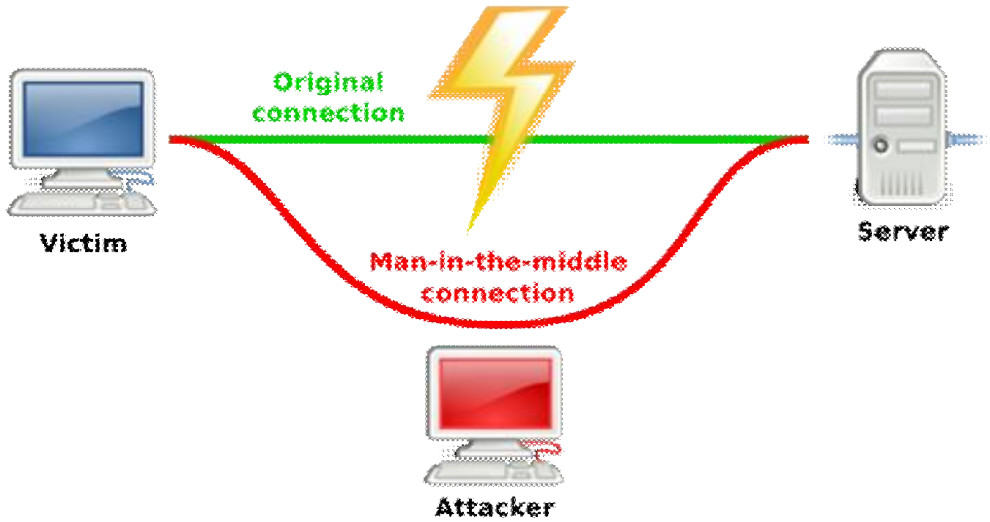
სოციალური ინჟინერია - ერთ-ერთი პოპულარული შეტევის გზა არის სოციალური ინჟინერია, რომლის დროს ჰაკერი ინფორმაციულ სისტემებთან წვდომას იღებს მოტყუებით პირადი ურთიერთობის დროს, ელექტრონული წერილებით ან ტელეფონურ საუბრებაში. სოციალური ინჟინერიის მთავარი იდეა მდგომარეობს იმაში, რომ ჩავსვათ ადამიანი ქსელის დარღვევის ციკლში და გამოვიყენოთ ის იარაღად. ინფორმაცია რომელსაც მსხვერპლი გასცემს შეიძლება გამოყენებული იყოს შემდგომი შეტევისთვის და სისტემაში შეღწევის ავტორიზაციის მიღებისთვის. მომხმარებელი ყოველთვის მოხსენიებული არის, როგორც ყველაზე სუსტი ბმული ქსელის უსაფრთხოებაში.

fishing შეტევა - fishing შეტევის დროს ჰაკერი ქმნის ყალბ ვებ გვერდს, რომელიც გამოიყურება ზუსტად როგორც რეალური პოპულარული საიტი, როგორც მაგალითად facebook-ის პირველი გვერდი. ამის შემდეგ ჰაკერი უგზავნის მომხმარებელს წერილს, რომელშიც არის ბმული ამ ყალბ გვერდზე. როდესაც user გადადის ამ საიტზე და შეიყვანს

პირად ინფორმაციას, ჰაკერი მიიღებს წვდომას ამ ინფორმაციაზე და ეცდება გამოიყენოს რეალურ საიტზე.

სნიფინგ (Sniffing) - ქსელური პაკეტების სნიფინგი არის ქსელში გამავალი მონაცემთა პაკეტების მოსმენა და აღება. სნიფერული პროგრამა მუშაობს ქსელის Ethernet დონეზე და იღებს მთლიან შემავალ და გადავალ ტრაფიკს. თუ Ethernet-ის პლატა ძებნის რეჟიმშია სნიფერის პროგრამა მიიღებს უფრო მეტ ინფორმაციას ტრაფიკიდან. სნიფერს, რომელიც დაყენებულია ქსელის ხერხემალ მოწყობილობაზე ან ქსელის აგრეგაციის წერთილზე საშუალება აქვს ქსელის მთლიანი ტრაფიკის მონიტორინგის. სნიფერების უმრავლესობა არის პასიური, ისინი პასიურად უსმენენ მოწყობილობის ქსელურ ინტერფეისში შემავალ და გამავალ მონაცემების პაკეტებს. ინტერნეტში არსებობს უამრავი სნიფერული პროგრამა. მათგან უფრო დახვეწილი უფრო აქტიური შეტევის საშუალებას გვაძლევს. სნიფინგიდან ყველაზე საუკეთესო დაცვაა end-to-end ან user-to-user ტრაფიკის შიფრაცია.

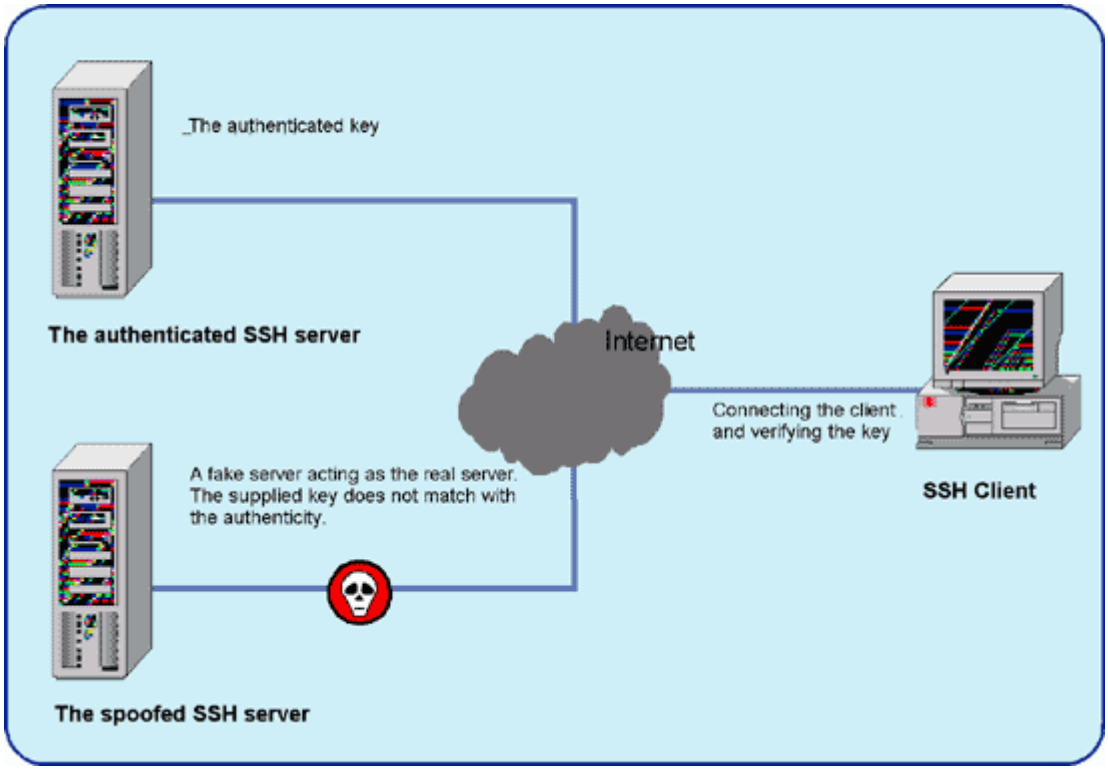
გატაცების შეტევა (man-in-the-middle attacks) - ეს ტექნიკა იყენებს TCP/IP პროტოკოლის არქიტექტურაში არსებულ სისუსტეებს. გატაცება ხდება როდესაც ვიღაც ერევა და აკონთროლებს თქვენი კომუნიქაციის პროცესის. როდესაც კომპიუტერები ურთიერთობენ ქსელის დაბალ დონეებზე, მათ შეიძლება ვერ დაადგინონ სწორად ვისთან ცვლიან მონაცემებს. თქვენ ფიქრობთ, რომ საუბრობთ ორიგინალურ პარტნიორთან, მაგრამ რეალორად ყველა პირად ინფორმაციას ხედავს ჰაკერი.



Spoof შეტევა - spoofing შეტევის დროს ჰაკერი ცვლის გამოგზავნილი მონაცემთა პაკეტების წყარო მისამართს ისე, რომ პაკეტები გამოჩნდეს, როგორც სხვა წყაროდან გამოგზავნილი.

ეს შეიძლება firewall წესების შემოვლითი გზებით გავლის მცდელობა იყოს. ქსელში ყველა ჩართული მოწყობილობა აუცილებლად აგზავნის IP მისამართებს ქსელში. ასეთი ინტერნეტ მონაცემების პაკეტები ინახავენ გამომგზავნის IP მისამართს და აპლიკაციის დონის მონაცემებს. თუ ჰაკერი მიიღებს კონტროლს ქსელში გაშვებულ პროგრამულ უზრუნველყოფაზე, ის ადვილად შეძლებს შეცვალოს მოწყობილობის პროტოკოლები იმისათვის, რომ განათავსო თვითნებური IP მისამართი მონაცემთა პაკეტის წყაროს მისამართის ველში. ეს ტექნიკა ცნობილია, როგორც IP spoofing, რომელსაც შეუძლია ყველა პაკეტის წყაროს მისამართის შეცვლა სხვა ნებისმიერ მისამართზე. პაკეტში შეცვლილი წყარო IP მისამართით ძნელია დავადგინოთ რეალურად ვინ გამოაგზავნა მონაცემები.

Spoofing-თან დაცვა არის ადრესების ფილტრაცია. ამის გაკეთება შეუძლია ყველა როუტერს. როუტერები ამოწმებენ IP მისამართიდან მიღებულ დატაგრამებს და განსაზღვრავენ არიან თუ არა მისამართები იმ მისამართებს შორის, რომლებიც არიან ინტეფეისით მისაწვდომი. თუ წყარო მისამართი გამომგზავნის პაკეტში არ არის დაშვებული სივრციდან, მაშინ ასეთ პაკეტებს როუტერი ბლოკავს.



Denial-of-Service შეტევა (DoS) - მომსახურებაზე უარის თქმის შეტევა არის სპეციალური ტიპის შეტევა, რომელიც მიზანად ისახავს დიდი საიტების გატეხვას. ამ ტიპის შეტევა ქსელში შექმნილია იმისთვის, რომ გამოიყვანოს ქსელი მწყობრიდან დიდი რაოდენობის უსარგებლო ტრაფიკის გამოგზავნით. მომსახურებაზე უარის თქმა ხდება როდესაც ისეთი სისტემა როგორიცაა ვებ სერვერი გადაივსება არალეგიტიმური მოთხოვნებით და ამით არ მისცემს მას საშუალებას ლეგიტიმურ მოთხოვნების უპასუხოს.

Buffer overflow – ბუფერის გადავსება არის, როდესაც ჰაკერი აგზავნის აპლიკაციაში უფრო მეტ მონაცემს ვიდრე ის ელოდებს. ბუფერის გადავსების შეტევა როგორც წესი, იწვევს სიტუაციას როდესაც თავდამსხმელი იძენს ადმინისტრაციულ უფლებებს shell-ზე.

Smurf შეტევა - ამ შეტევის დროს თავდამსხმელი აგზავნის IP პინგის მოთხავნებს მიმღებ საიტზე. ping პაკეტი აღნიშნავს, რომ ის მიმართულია რამოდენიმე ჰოსტზე სისტემის შიგნით. პაკეტი ასევე აღნიშნავს, რომ ის არის მოთხოვნა რაიმე სხვა საიტიდან, რომელიც არის მომსახურებაზე უარის თქმის შეტევის სამიზნე. შეტევის შედეგად სამიზნე საიტი მიიღებს დიდი რაოდენობის პასუხებს, რომლებსაც სწორად ვერ დაამუშავებს და თუ მიიღებს საკმარისად ბევრ პასუხს, ჰოსტი შეძილება გამოვიდე მწყობრიდან და ვერ მიიღოს რეალური ტრაფიკი.

SYN floods – როდესაც კომპიუტერი ამყარებს კავშირს სხვა კომპიუტერთან, როგორც წესი სერვერთან ხდება TCP/SYN და TCP/ACK ინფორმაციის პაკეტების გაცვლა. კომპიუტერი რომელიც ითხოვს კავშირს (კლიენტის ან მომხმარებელის კომპიუტერი), აგზავნის TCP/SYN პაკეტს, რომელიც უგზავნის დაკავშირების მოთხოვნას სერვერს. თუ სერვერი მზად არის კავშირის დასამყარებლად ის უგზავნის TCP/SYN-ACK პაკეტს უკან კლიენტს პასუხით „დიახ, კავშირი შესაძლებელია“, არეზერვირებს ადგილს კავშირისათვის და ელოდება კლიენტის TCP/ACK პაკეტს. SYN flood-ში კლიენტის მისამართი შეცვლილია ასე, რომ სერვერი უგზავნის კლიენტს TCP/SYN-ACK პაკეტს, მაგრამ მაგისი შეტყობინება არ არის მიღებული, რადგან კლიენტი არ არსებობს ან არ ელოდება რაიმე შეტყობინებას და აიგნორებს გამოგზავნილ პაკეტს. ეს ტოვებს სერვერს მკვდარი კავშირით, რომელიც დაზერვირებულია კლიენტის პასუხისთვის, რომელიც თავის მხრივ არასდროს არ მოვა. როგორც წესი, ეს ოპერაცია გამოყენებულია ბევრჯერ იმისათვის, რომ სერვერმა დაარეზერვიროს ადგილი ყველა ამ კავშირისათვის და როდესაც არ დარჩება ადგილი

კავშირის რეზერვირებისათვის, ლეგიტიმური კლიენტები ვერ დაამყარებენ ახალ კავშირებს.

Distributed-Denial-of-Service (DDoS) - DDoS შეტევა ხდება როდესაც რამოდენიმე გატეხილი სისტემა ან რამოდენიმე ჰაკერი ერთროულად აკეთებს ბევრ მოთხოვნას სერვერზე და უაზრო ტრაფიკით ბლოკავს სერვისს. DDoS-ის დროს ჰაკერმა ჯერ უნდა მიიღოს წვდომა დიდი რაოდენობის ინტერნეტ ჰოსთებთან. ამის მერე ის აყენებს ამ ჰოსტებზე შემტევ პროგრამას, რომელიც მშვიდად ელოდება ბრძანებას control პროგრამიდან, რომელსაც აქვს საშუალება დაუკავშირდეს ყველა ჰოსტზე დაყენებულ პროგრამას, მიუთითოს შეტევის სამიზნე და ერთდროულად ამ სამიზნეზე გაუშვას შეტევა. შედეგად კოორდინებული შეტევა განსაკუთრებით ზიანისმომტანია, რადგან ერთროულად მოდის ბევრი ჰოსტიდან. როუტერების აქვთ წვდომის ფილტრი, რითაც შეუძლიათ DoS შეტევის ფილტრაცია და ისიც მცირე მოცულობით, ამიტომ DDoS არის ერთ-ერთი ყველაზე მარტივი და პოპულალური შეტევის ტიპი.

Exploit შეტევა – ამ შეტევის დროს ჰაკერმა იცის უსაფრთხოების პრობლემის შესახებ ოპერაციულ სისტემაში ან პროგრამული უზრუნველყოფაში და ამ პრობლემის გამოყენებით ტეხვს სისტემას.

ტროიანები – ეს პროგრამები გამოიყურება, როგორც ჩვეულებრიბრივი პროგრამული უზრუნველყოფა, მაგრამ რეალურად ასრულებენ გაუთვალისწინებელ ან თავდამსხმელის ქმედებებს გაშვების დროს. დისტანციური მართვის spyware პროგრამები ძირითადად ამ ტიპის არიან. trojan გამოყენების ტექნიკების რაოდენობა შეზღუდულია მხოლოდ თავდამსხმელის ფანტაზიით. დავირუსებული ფაილი ჩანს, იგივე ზომის როგორც რეალური ფაილი. ერთადერთი ეფექტური დაცვა არის კრიპტოგრაფიული ჯამის ან ორობითი ციფრული ხელმოწერის დროული გამოყენება.

პაროლიანი შეტევა – ამ შეტევის დროს ჰაკერი ცდილობს პაროლების გატეხვას, რომლებიც შენახულია ქსელის აკაუნთების მონაცემთა ბაზაში ან დაპაროლებულ ფაილში. არსებობს ამ შეტევის სამი ძირითადი ტიპი: ლექსიკონიანი შეტევა, brute-force შეტევა და ჰიბრიდული შეტევა. ლექსიკონიანი შეტევა იყენებს სიტყვების სია ფაილს, რომელშიც ჩაწერილია სავარაუდო პაროლები. Brute-force შეტევის დროს ჰაკერი ცდილობს ყველა შესაძლო სიმბოლოს კომბინაციით იპოვოს სწორი პაროლი.

SQL injection შეტევა - **injection**-ს დროს ჰაკერი სვამს კოდს სერვერის მონაწემთა ბაზის SQL მოთხოვნაში. კოდი აფუჭებს საიტის რაიმე ველს, რომლის მანაცემები ჩაწერილი უნდა იყოს ბაზაში. წარმატებულ SQL injection-ს შეუძლია წაიკითხოს მნიშნელოვანი ინფორმაცია მონაცემთა ბაზიდან, შეცვალოს ბაზის მონაცემები, შეასრულოს ადმინისტრაციული ოპერაციები მონაცემთა ბაზაზე (მაგალითად DBMS-ის გათიშვა), გაიგოს DBMS ფაილური სისტემის ფაილების შინაარსი და ზოგიერთ შემთხვევაში შეასრულოს ბრძანებები ოპერაციულ სისტემაში.

უსაფრთხოების აუდიტის ტიპები

გარე დაუცველობის შეფასებები – ამ შეფასების ტიპი აიდენტიფიცირებს და ამოწმებს ქსელის ინტერნეტში ჩართულ მოწყობილობებს უსაფრთხოების ხარვეზებს გარე შეტევებიდან და არის თუ არა საშუალება ამ ხარვეზების გამოყენება იმისათვის, რომ გატეხოს სამიზნე სისტემა ან წვდომა მიიღოს მნიშნელოვან ინფორმაციაზე. ასევე ამ შეფასების ეტაპზე შესაძლებელია გაკეთდეს ნებაყოფილობითი გარე შეღწევადობის ტესტირება იმისათვის, რომ რეალურად შემოწმდეს ქსელის უსაფრთხოების ხარვეზები.

შინაგანი დაუცველობის შეფასებები - ამ შეფასების ტიპი ამოწმებს ქსელს და მასში ჩართულ მოწყობილობებს შესაძლებელია თუ არა მათი სისუსტის გამოყენება იმისათვის, რომ მიღებულიქნეს არასანკციონური წვდომა ინფორმაციასთან ქსელის შიდა შეტევის დროს. ამ შემოწმების დროს ასევე შესაძლებელია შეღწევადობის ტესტირების გაყეთება.

ქსელის არქიტექტურის მიმოხილვა - ქსელის არქიტექტურის მიმოხილვა აფასებს არსებული უსაფრთხოების კონტროლების და ქსელურ მოწყობილობების ფუნქციას, განთავსება და ხარვეზები და ადერების გამოყენებას, ორგანიზაციის უსაფრთხოების მიზნების და ამოცანების ფარგლებში.

უსადენო უშიშროების მიმოხილვა - უსადენო შეღწევადობის ტესტირება და შეფარსების სერვისები ამოწმებენ უსადენო ქსელის ინპლიმენტაციის უსაფრთხოებას და ტესტირების შემდეგ გვიწევენ რეკომენდაციას გაუმჯობესებისთვის.

VPN უსაფრთხოების მიმოხილვა - VPN უსაფრთხოების განხილვა ადარებს ქსელის არსებულ VPN (Virtual Private Network) კონფიგურაციას რეკომენდირებულ პრაკტიკებთან და განსაზღვრავს მასში რაიმე შეშფოთების მომენტებს. ასევე განხივის დროს ხდება დისტანციური და ადგილობრივი კონფიგურაციის განხილვა და არქიტექტურის მიმოხილვა.

Firewall უსაფრთხოების მიმოხილვა - Firewall უსაფრთხოების მიმოხილვის დროს ტესტრდება firewall-ის კონფიგურაცია და შესაძლებელი შეტევებისგან დაცვის ეფექტურობა. Firewall უსაფრთხოების მიმოხილვები ძალიან მნიშნელოვანია, რადგან ისინი აიდენტიფიცირებენ ხარვეზებს, რომლებიც შეიძლება არ დაფიქსირდეს ქსელის შეღწევადობის ტესტებისა და „შავი ყუთი“ (black box) ტიპის ქსელის შეფასებების დროს.

მობილური მოწყობილობის მიმოხილვა - ამ შეფასების მიზანი არის პოტენციური უსაფრთხოების ხვრელების გაგება მობილური ტექნოლოგიის გამოყენების დროს ქსელში, მაგალითად, მობილურიდან ქსელში შიდა და გარე შესვლისას.

Active Directory-ს მიმოხილვა - Active Directory პასუხს აგებს მომხარებელთა დაცვის უფლებების კონტროლზე და რადგან ის ასეთ კრიტიკულ როლს თამაშობს ქსელის გარემოში, მისი უსაფრთხოების მიმოხილვა არის ძალიან მნიშნელოვანი და საშულებას გვაძლევს აღმოვაჩინოთ ისეთი ხარვეზების, რომლებიც ნათლად არ ჩანდნენ ქსელის ტესტირების დროს.

შეფასების სერვისების განმარტებები

უსაფრთხოების პროგრამული უზრუნველყოფის მწარმოებლები გთავაზობენ დიდი რაოდენობით სხვადასხვა გზებით ბრენდირებული უსაფრთხოების შეფასებების გზებს. ქვედა ნახაზი გვიჩვენებს მთავარ სერვისებს, შეფასების სიღრმესა და შედარებით ღირებულებას. თითოეული მომსახურების ტიპს შეუძლია სხვადასხვა უსაფრთხოების ხარისხის უზრუნველყოფა.

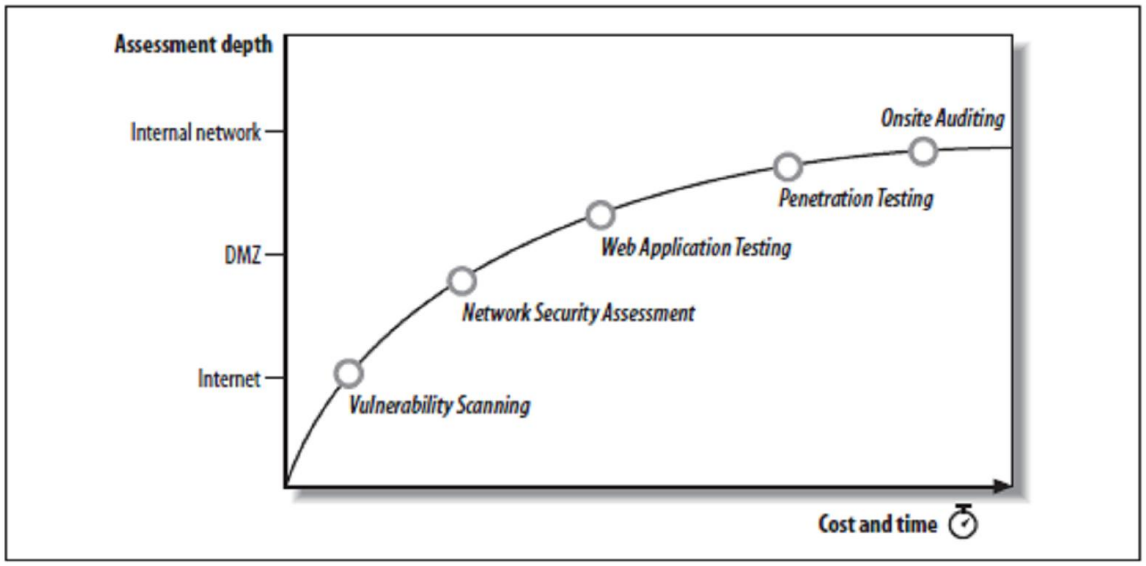


Figure 1-1. Different security testing services

დაუცველობის სკანირება - დაუცველობის სკანირება იყენებს ავტომატურ სისტემებს (მაგ. Nessus, ISS Internet Scanner, QualysGuard, eEye Retina) რომლის გამოყენება შესაძლებელია მინიმალური ქსელის უსაფრთხოების შეფასების პრაქტიკული და კვალიფიკაციური ცოდნის გარეშე. ეს არის იაფი გზა დასაზუსტებლად, რომ არ არსებობს აშკარა ხარვეზი ქსელის უსაფრთხოებაში, მაგრამ ეს მეთოდი არ გთავაზობს ქსელის უსაფრთხოების გაუმჯობესების მკაფიო სტრატეგიას.

ქსელის უსაფრთხოების შეფასება - არის ეფექტური ნაკრები ავტომატური და პრაქტიკული დაუცველობის ტესტირებისა და შეფასების. ამ ტესტირების შედეგი, როგორც წესი, არის მოხსენება, ხელნაწერი, ზუსტი და ლაკონური და იძლევა ქსელის გაუმჯობესების პრაქტიკულ რჩევებს.

ვებ აპლიკაციის ტესტირება - მოიცავს ავტორიზაციის გასვლის შემდეგ ვებ აპლიკაციის კომპონენტების შეფასებას, არასანქცირებული ინიექცირებული ბრძანებების იდენტიფიკაციას, ცუდად განაწილებულ უფლებებს და სხვა უსაფრთხოების სისუსტეებს მოცემულ აპლიკაციაში. ამ დონის ტესტირება მოიცავს ვრცელ საკვალიფიკაციო ხელნაწერ ტესტირებას და კონსულტანტის ჩართულობას საქმეში და ამიტომ მისი ავტომატიზაცია ძნელია.

შეღწევადობის ტესტირება (penetration testing) - ქსელის სრული შეღწევადობის ტესტირება მოიცავს მრავალ თავდასხმის ვექტორს (მაგ., ტელეფონით "war dialing", სოციალური ინჟინერიას და უკაბელო ტესტირებას) იმისათვის, რომ შეასრულოს შეტევა სამიზნე გარემოზე. ეს ტესტირება მოიცავს ჰაკერების მიერ რეალური მეთოდების გამოყენებას და ამიტომ კარგად აჩვენებს უსაფრთხოების სუსტ ადგილებს.

ადგილზე აუდიტი - გთავაზობს ქსელის უსაფრთხოების ყველაზე ნათელ სურათს. უსაფრთხოების კონსულტანტებს აქვთ ადგილობრივი სისტემთან სრული წვდომა და შეუძლიათ გაუშვან უსაფრთხოების შეფასების ინსტრუმენტები თითოეულ სისტემაში და შეუძლიათ გაიგონ დაცვის ყველა პრობლემა, ისეთი როგორც rootkit-ის გამოყენება, სუსტი მომხმარებლის პაროლები, ცუდად განსაზღვრული უფლებები და სხვა სახის უსაფრთხოების დარღვევებს. 802.11 უსადენო ტესტირება ასევე ხშირად გამოიყენება ადგილობრივი აუდიტის დროს.

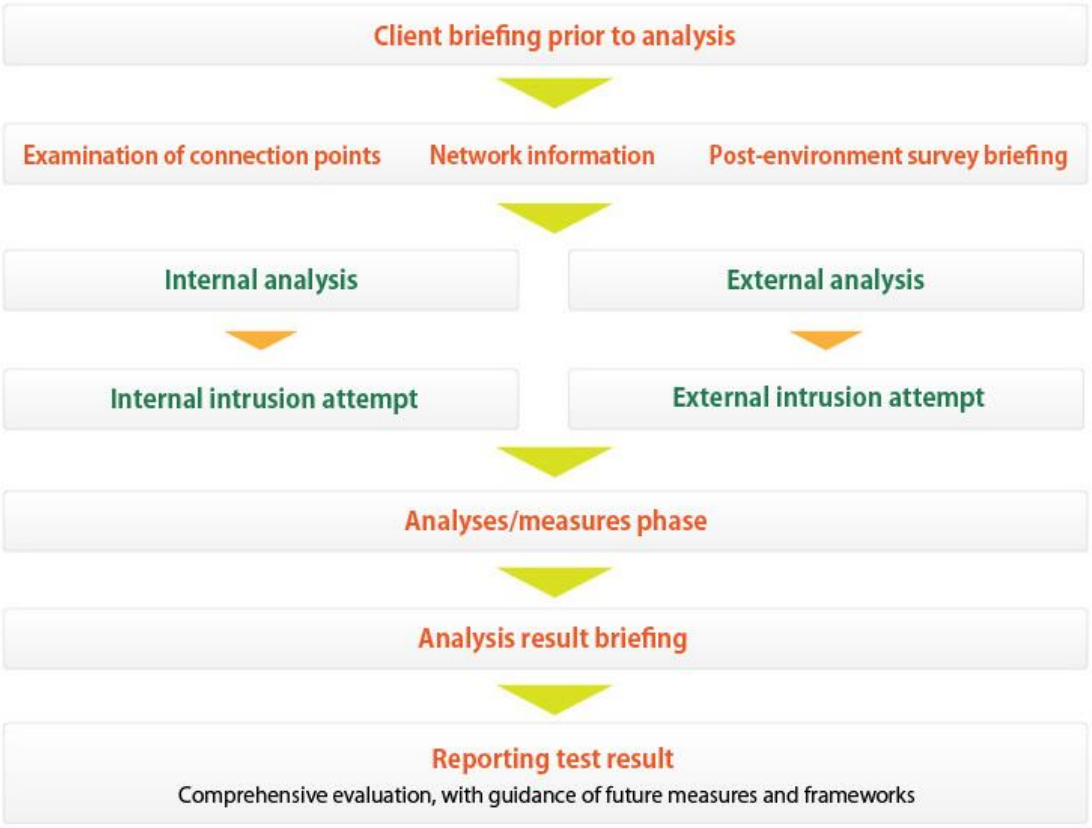
უსაფრთხოების შეფასების ტიპიური ნაბიჯები ქსელის შემოწმების დროს

ქსელის უსაფრთხოების შეფასების დროს რისკის დონე ფასდება ეტაპობრივად. ჯერ შემფასებელმა ქსელში წვდომა უნდა მიიღო ინტენეტით ან intra-LAN-ით, ამის შემდეგ იგი ცდილობს აღმოაჩინოს სისუსტეები თითოეული ტერმინალში (host/node), რომელიც ხელმისაწვდომია მოცემულ ქსელში და მერე უშვებს იმიტირებული შეტევებს.

შეფასების მახასიათებლები

ყველა ჰოსტი, რომელიც ხელმისაწვდომია ანალიზში მოწმებდება შიდა-ჰოსტის რისკების არსებობაზე. დატესტილი უნდა იყოს ყველა გარე საფრთხე, მათ შორის, ინტერნეტიდან შემასული ტრაფიკი, კავშირი პარტნიორებს შორის და უსადენო კავშირები. ამის შემდეგ უნდა გავუშვათ თავდასხმები ადმინისტრატორის გაფრთხილების გარეშე და ეს თავხდასხმა უნდა გაიფანტოს დომინო-ეფექთით ყველა ჰოსტზე.

ქსელური უსაფრთხოების შეფასების ეტაპები



კლიენტის ბრიფინგი ანალიზის დაწყებამდე

ამ ბრიფინგზე დადასტურება ანალიტიკური პროცედურები და საკომუნიკაციო / კოორდინაციის საკითხები კლიენტთან და ანალიტიკურ ექსპერტების შორის. შეტევის დროს ასევე უნდა აღმოაჩინო კლიენტის ქსელის ანომალიის გამოვლენის შესაძლებლობა და რამოდენიმე ტესტირების ეტაპი განხორციელი უნდა იყოს კლიენტის ცოდნის გარეშე. ამიტომ კლიენტის მიერ ქსელში შეტევის აღმოჩენის დროს გაწერილი უნდა იყოს შესაბამისი კომუნიკაციის პროცედურები.

დიაგნოსტიკური ეტაპი - 1: გარემოს გამოკვლევა

ამ ანალიზის ეტაპზე იკვლევა კავშირების წერტილები და მათი რაოდენობა კლიენტის ქსელში. შემოწმებულია ინტერნეტთან დაკავშირებული ყველა მოწყობილობა, პარტნიორებს შორის კერძო ქსელების კავშირები და უსადენო LAN წვდომის წერტილები. ამ საჯარო და კერძო კავშირების და მოწყობლობის სია მზადდება და იგზავნება კლიენტთან იმისათვის, რომ კლიენტმა შეძლოს გაიგოს რა კავშირები გაიტესტება და დაადასტუროს ამ კავშირების რეალური არსებობას.

როდესაც კლიენტის ქსელის ინფორმაცია შეგროვებულია ანალიზი გადადის შემდეგ ეტაპზე. ვიდრე ეს ეტაპი დაიწყება შეიძლება დასჭირდეთ კიდევ ერთი ბრიფინგი კლიენტთან. მაგალითად, თუ ტესტერის მიერ აღმოჩენილი ტოპოლოგია და კავშირების წერთილები არ ემთხვევა კლიენტის მიერ მოწოდებულ ინფორმაციას ტოპოლოგიის შესახებ (მაგ. არადოკუმენტირებული ჰოსტების და დაკავშირების წერთილების აღმოჩენის დროს), შეიძლება დასჭირდეს შეხვედრა ქსელზე პასუხისმგებელ პირთან. ეს საჭიროა იმისათვის, რომ გააცნოთ კლიენტს კვლევის შედეგი და ზუსტად განსაზღრული და შეთანხმებული იყოს ქსელის ტესტირების სივრცე.

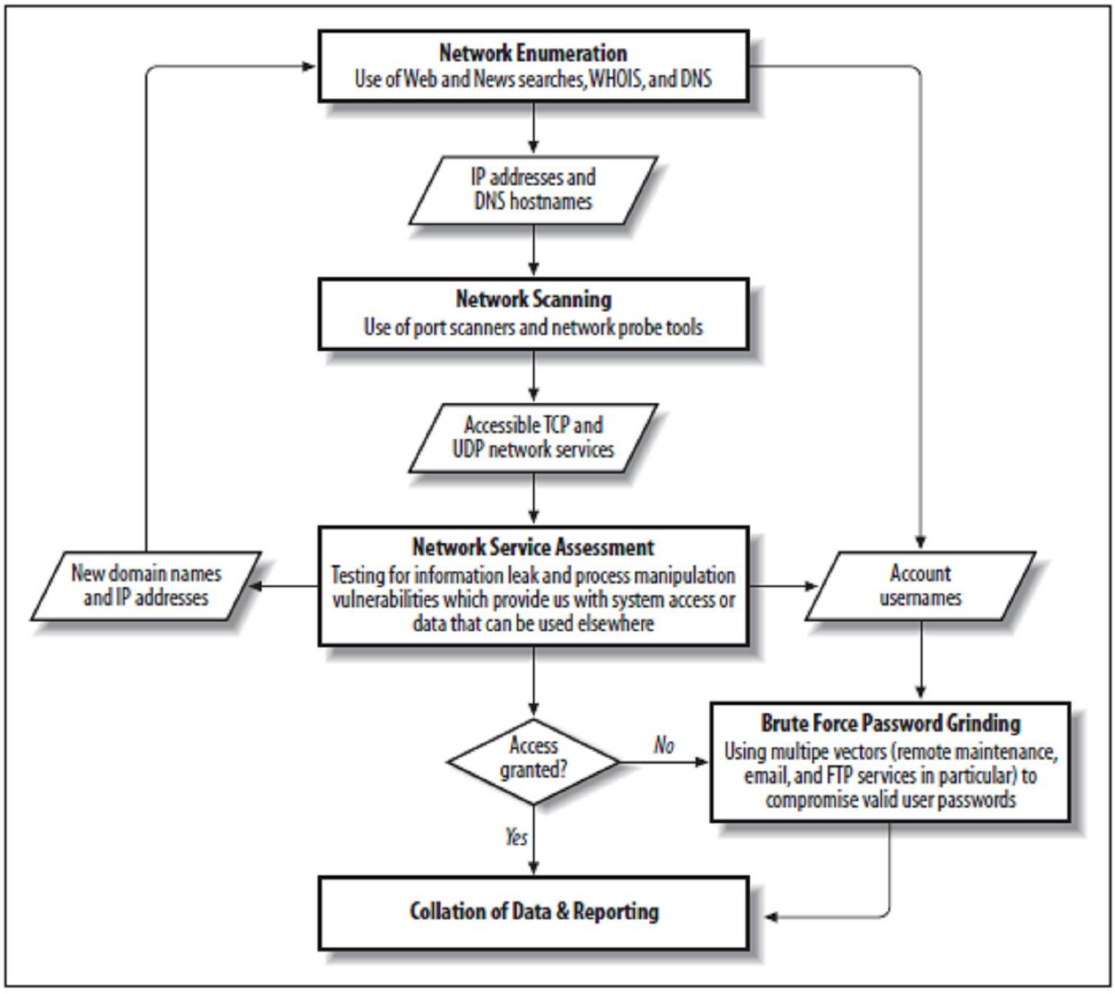


Figure 1-2. The cyclic approach to network security assessment

დიაგნოსტიკური ეტაპი - 2: ანალიზი

შეტევის ეტაპი, რომელიც იწყება ქსელის ანალიზის დროს მიმდინარეობს ერთდროულად ქსელის შიგნიდან და გარედან და ინიციალიზირებულია ცალ-ცალკე. გარე ანალიზი მიმართულია დაუცველობის აღმოჩენისაკენ კლიენტის სისტემაში. მეორე მხრივ შიდა ანალიზი ყურადღებას აქცევს super-user-ის დაუცველობებზე, როდესაც ჰაკერი

ავტორიზებულია სისტემაში მაოხმარებელის უფლებებით. ეს ორმხრივი ანალიზი დაკავშირებულია. როდესაც თავდამსხმელი გარედან შეძლებს შეტევის განხორციელებას, იგი შეძლებს შიდა სისუსტეების გამოიყენებას რამაც შესაძლებელია გამოიწვიოს მორიგი დაზიანების გაფართოება, რომელიც ცნობილია, როგორც „დომინოს ეფექტი“. ანალიზის დროს ტესტირების ორი მხრივი შეტევების შეგედები გაერთიანებულია იმისათვის, რომ განსაზღვრებული იყოს კლიენტის ქსელის მთლიანი მოწყვლადობა. ასევე ამ ეტაპის შედეგები დამატებული უნდა იყოს საბოლო მოხსენებაში.

„დომინოს ეფექტი“

თავდამსხმელი რომელმაც წვდომა მიიღო მოწყობილობაზე, რომელიც პირველი შეხედვით არამნიშნელოვანია სესტემისთვის, შეიძლება ჯაჭვითი შეტევებით გაფართოვოს მისი კონტროლი სხვა მოწყობილობებზე, რომელშიც ინახება მნიშნელოვანი ინფორმაცია. თავდამსხმელი ჯაჭვითი შეტევებით საბოლოოდ აღწევს მის მიზანს - სისტემა და მისი დომეინი. დომინოს ეფექტი არის ისეთი სიტუაცია, როდესაც ჰაკერს შეუძლია გამოიყენოს ასეთი ჯაჭვური ხარვეზები.

მაგალითად ასეთი სიტუაცია: ქსელში გამოყენებული იყო გარე ქსელიდან მისაწვდომი პრინტერი, პრინტერებს არ აქვთ მომხმარებელთა აქაუნთი ამიტომ ადვილად შესაძლებელი იყო მასთან დაკავშირება. ამ პრინტერზე SNMP (Simple Network Management Protocol) პროტოკოლი იყო გაშვებული. პრინტერის კონფიგურაციის ფაილს დათვალიერებამ საშუალება მისცა თავდასხმელს გაეგო ქსელური მოწყობილობების სახელები და სხვა მნიშვნელოვანი ინფორმაციას. SNMP მიღებული ინფორმაციის საფუძველზე, ჰაკერმა შეძლო Cisco როუტერში შესვლა. Cisco როუტერის კონფიგურაციის ფაილის ნახვამ მიეცა შესაძლებლობა მიეღო ინფორმაციას აქაუნტზე, აქაუნტის სახელი და ჰეშირებული პაროლი. პაროლების ანალიზის შემდეგ წარმატებულად დასრულდა DC (Domain Controller) სერვერზე ავტორიზაციის მცდელობა. DC-ის ადმინის უფლებებით დომენის ჰოსტის ყველა SAM ფაილი მიღებული იყო. შედეგად თითქმის ყველა დაკავშირებული ჰოსტი დაპყრობილიქნა.

დიაგნოსტიკური ეტაპი - 3: თავდასხმა

ქსელის ტესტირება რეალურ თავდასხმაზე საშუალებას გვაძლევს ისეთი უსაფრთხოების ხარვეზების გაგებას, რომლებიც არ ფიქსირდება ჩვეულებრივი დიაგნოსტიკური ინსტრუმენტების გამოყენების დროს. ავტომატური დიაგნოსტიკის ინსტრუმენტები

ხშირად პოულობენ უსაფრთხოების ხარვეზებს, მაგრამ არ ამოწმებენ არის თუ არა ამ დაუცველობებზე რეალური საფრთხე.

უფრო მეტიც ასეთი ინსტრუმენტები ხშირად მუშაობენ მარტო TCP/IP ქსელის ფარგლებში. თავდასხმის ეტაპზე უსაფრთხოების ანალითიკოსი ხელით ასრულებს შეტევას კლიენტის ქსელზე.

ის ცდილობს არასანქცირებული წვდომის მიღებას, როგორც შიდა მომხმარებელი და ასევე სიმულირებს შეტევა ინტერნეტიდან, მონაცემთა ქსელიდან, უსადენო კავშირიდან და ამით ამოწმებს შიდა და გარე უსაფრთხოების ხარვეზებს.

შეტევის აღმოჩენის სისტემა კლიენტის ქსელში და ასევე მომხმარებლებისა და სისადმინის ანომალიების გამოვლენის უნარი ტესტირდება სიმულაციური შეტევის დროს. ამიტომ ტესტირება ჯერ უნდა იყოს გაშვებული ძნელად შესამჩნევი მეთოდებით და მერე ნელნელა შეიცვალოს მარტივად შესამჩნევ მეთოდებზე.

ანალიტიკურმა ექსპერტმა ასევე უნდა გაითვალისწინოს საწყისი სკრინინგის შედეგები და ანალიზის ეტაპზე დადასთურებული მოწყვლადობები იმისათვის, რომ სხადასხვა გზით შესრულებული შეტევებით აღმოაჩინოს რეალური საფრთხეები, რომლებიც დამალული შეიძლება იყონ კლიენტის ქსელში.

შეტევის ეტაპი ხდება ხოლმე კლინტის ნებით თუ მას სჭირდება სისტემის ღრმა ანალიზი, დანარჩენ შემთხვევაში ხდება მარტო სისტემის ანალიზი შეტევის ეტეპის გარეშე.

დიაგნოსტიკური ეტაპი - 4: მოხსენება

ამ ეტაპზე ხდება ყველა ეტაპების შედეგების შეგროვება, შეხვედრა კლიენტებთან და მისთვის ამ შედეგების გაცნობა, იმისათვის, რომ შეთავაზებული იყოს ქსელის უსაფრთხოების გაუმჯობესების შესაძლებლობები. ასევე ხდება ფინალური მოხსენების ფრომირება, რომელშიც გაწერილია კონკრეტული უსაფრთხოების ხარვეზები, არასაიმედო პრაქტიკები, პარამეტრები, ადმინისთრაციის და ქსელის არქიტექრურის შეცდომები და ასევე რეკომენდაციები ყველა ამ საკითხზე.

ასევე შეილება გაიმართიოს სპეციალური ბრიფინგი კლიენტთან, რომელშიც განხილული იქნება ანალიზის ძირითადი შედეგები, გაიმართება პრეზენტაცია გამოყენებული შეტევის მეთოდებზე და ასევე კლიენტს შეეძლება შეკითხვებს დასმა უსაფრთხოების ანალიტიკოსებისთვის.

მოხსენების შინაარსი

საბოლო მოხსენებაზე გაწერილი უნდა იყოს ანალიზის ყველა შედეგი, ისეთი როგორც: შესრულების რეზიუმე, ანალიზის მიზანი, გარე და შიდა დაუცველობებზე რეზიუმებული ინფორმაცია, ანალიზის პროცედურა, გარემოს კვლევის ფაზის შედეგი, შეტევის ეტაპის შედეგი, რეკომედაციები უსაფრთხოების გაუჯობესებისთვის, ქსელის დეტალური რუკა, კონფიგურაციის და ადმინისტრირების რეკომენდაციები, მომხმარებლების ავტორიზაციის შესახებ ინფორმაციის ადმინისტრირების პირობები და რეკომენდაციები, ქსელის დიზაინის აღწერა და გაუმჯობესების შესაძლებლობა, DNS ზონებზე ინფორმაცია, შეღწევადობის შედეგები და ა.შ.

ტესტირების ინსტრუმენტების მიმოხილვა

კარგი დაუცველობის შეფასების ინსტრუმენტის თვისებები

ვიდრე დავიწყებთ ინსტრუმენტებისა და მათი შესაძლებლობების განხილვას, უნდა განვმარტოთ რითი გამოირჩევა კარგი მოწყვლადობის შეფასების ინსტრუმენტი. მიუხედავად იმისა რა ტიპის ინსტრუმენტი გამოიყენება, კარგ ინსტრიმენტს მინიმუნ უნდა ჰქონდეს შემდეგი მახასიათებლები:

ცრუ-დადებითი (false positive) შედეგების მცირე რაოდენობა.

ახალი მოწყვლადობის შეფასების ინსტრუმენტის შექმნის დროს ერთ-ერთი გამოწვევა, რომელიც დეველოპერის წინაშე დგას არის ცრუ-დადებითი შედეგი. ცრუ-დადებითი შედეგი ხდება, როდესაც ინსტრუმენტი აიდენტიფიცირებს პრობლემას რომელიც რეალურად არ არსებობს, ან არასწორად აიდენტიფიცირებს რეალურ პრობლემას, როგორც არამნიშნელოვანს. მიუხედავად იმისა, რომ საკამათოა, შეუძლია თუ არა ანალიზის ინსტრუმენტს მთლიანად გაფილტროს ცრუ-დადებითი შედეგები, მაგრამ ასეთი შემთხვევების დიდი რაოდენობა მიუღებელია, რადგან ამან შეიძლება გამოიწვიოს პრობლემა დიდი საწარმო ქსელების ტესტირების დროს.

ნულოვანი ცრუ უარყოფით შედეგები

ყველაზე ცუდი რაც შეიძლება მოხდეს მოწყვლადობის შეფასების ინსტრუმენტთან, მოწყვლადობის პოვნის უუნაროობა. ამ სიტუაციას ეწოდება ცრუ უარყოფითი.

მოწყვლადობის არ აღმოჩენის დროს არამარტო სისტემა რჩება დაუცველი არამედ მომხმარებელს აქვს უსაფრთხოების განცდის ცრუ გრძნობა.

მოკლე და სრული შემოწმების მონაცემთა ბაზა

ეს არის ერთ-ერთი სივრცე რომელშიც უსაფრთხოების პროგრამული უზრუნველყოფის მწარმოებლები თამაშობენ ერგეთწოდებულ ციფრების თამაშს. ერთ-ერთი პრობლემა მოწყვლადობის შეფასების სფეროში არის დასახელების სტანდარტების ნაკლებობა ხარვეზებისთვის, რაც გვაძლევლევს საშუალებას უსაფრთხოების პროგრამული უზრუნველყოპის მწარმოებლებს დაასახელონ და დათვალონ უსაფრთხოების ხარვეზები ისე, როგორც მოუნდებათ. მაგალითად, მწარმოებელი A იძახის, რომ მისი უსაფრთხობის ინსტრიმენტს შეუძლია დააფიქსიროს 1400 ხარვეზი და მწარმოებელი B თავის მხრივ გვეუბნება, რომ მისი ინსტრუმენტით შესაძლებელია 2000 სხვადასხვა ხარვეზის დაფიქსირება. ნიშნავს თუ არა ეს იმას, რომ B გამყიდველის ინსტრუმენტი რეალურად ამოწმებს უფრო მეტი უსაფრთხოების ხარვეზს, თუ უბრალოდ სხვაგვარად ითვლის ამ ხარვეზებს?

მაგალითად, საერთო მოწყვლადი და გამოვლინებების მონაცემთა ბაზა (Common Vulnerabilities and Exposures (CVE) database) რომელიც შექმნილია Mitre Corp. -ს მიერ, დიდი გზა გაიარა ამ პრობლემის მოგვარებისთვის, მაგრამ უსაფრთხოების სკანირების მწარმოებლები უბრალოდ ამატებენ CVE მითითებას მის შემოწმებაზე და მაინც აგრძელებენ მის დათვლას თავიანთი გზით.

მაგალითად, MS06-001 დაუცველობა ოპერაციული სისტემის გრაფიკულ ძრავში დისტანციურად კოდის გაშვების საშუალებას გვაძლევს სისტემაში და ამ დაუცველობას ერთადერთი მინიშნება აქვს CVE-ში (CVE-2005-4560). მაგრამ თუ წავიკითხავთ მწარმოებელთა რჩევას ამ ხარვეზზე (https://technet.microsoft.com/en-us/library/security/ms06-001.aspx) ჩვენ ვნახავთ, რომ ეს ხარვეზი შვიდ ოპერაციული სისტემაში არსებობს. და როდესაც უსაფრთხოების უზრუნველყოფის მწარმოებელი ხარ ცხადია, რომ არსებობს მკაფიო მარკეტინგული მიზეზი იმისათვის, რომ დაითვალოთ ეს ხარვეზი, როგორც შვიდი დაუცველობის შემოწმება ვიდრე, როგორც ერთი, და ეს არის ზუსტად ის რასაც ბევრი მოვაჭრე აკეთებს. ყველაზე კარგი ვარიანტია ამ შემთხვევაში შეადაროთ თუ ინსტრუმენტი ზუსტად შეესაბმება Mitre-ს CVE ბაზას (http://cve.mitre.org).

რწმუნებით (credentials) შემოწმება

დაუცველობის შეფასების ინსტრუმენტების ადრინდელ დღეებში სისტემის რწმუნებულების სკანირების კონცეფცია არ იყო განხილული. პირველი უსაფრთხობის

ინსტრუმენტების მწარმოებლები ამბობდნენ, რომ მათი პროდუქცია საშუალებას იძლევა შევხედოთ სისტემას თავდამსხმელის მხარიდან. რეალობაში სისტემაში არსებობენ, როგორც გარე ისევე შიდა უსაფრთხოების საფრთხეები და ამიტომ რწმუნებათა სიგელების არსებობა სისტმაზე სკანირების დროს ეხმარება ხარვეზების პოვნას. ასევე სისტემაში უფლებამოსილობის მინიჭება ეხმარება უფრო ზუსტი სკანირების შედეგების მიღებას, რადგან სკანირების ინსტრუმენტს ამ შემთხვევაში აქვს საშუალება შეამოწმოს სისტემის პარამეტრები და მაგალითად, რეგისტრის ჩანაწერებისა და ფაილების ვერსიებს. ყველა ასეთი შემოწმებისათვის სკანირების პროგრამას უნდა ჰქონდეს შესაბამისი უფლებები სისტემაში.

არა-უფლებამოსილი შემოწმება

მიუხედავად იმისა, რომ უფლებამოსილი შემოწმება მნიშნელოვანია სიზუსტისათვის, უფლებების გარეშე შემოწმებები არის ასევე მნიშნელოვანი, რადგან ეხმარებიან რეალური გარე საფრთხეების აღმოჩენისათვის. სისტემის რისკის შეფასების დროს მნისნელოვანია იმის გათვალისწინება, როგორ შეიძლება სისტემის კომპრომენტირება. შემოწმებები რომლებიც აბრუნებენ შედეგს ფართო სისტემის უფლებების გამოყენების გარეშე, ნამდვილად აჩვენებენ იმას რასაც დაინახავს ჰაკერი, რომელსაც ასევე არ აქვს ადმინისტრატორის უფლებები სისტემაში. ასეთი შემოწმებები ძნელია მოწყვლადობის შეფასების ინსტრუმენტისათვის, ამიტომ ეს ფუნქცია არის ინსტრუმენტის კარგი მაჩვენებელი.

დაბალი გავლენა ქსელის ტრაფიკზე

ყველა ვინც დიდი ხანი მუშაობდა მოწყვლადობის შეფასების სფეროში მიჩვეულია ტესტირების გაშვებაზე ღამის საათებში, როდესაც ქსელის ტრაფიკი დაბალია, რადგან ძველი მოწყვლადობის შეფასების ინსტრუმენტებს დიდი გავლენა ჰქონდათ ქსელის ტრაფიკზე და შესაბამისად ქსელის სიჩქარეზე. წლების განმავლობაში, უმრავლეს ინსტრუმენტებში ეფექტურობა და საიმედოობა გაუმჯობესდა, იმისათვის, რომ ტესტირება შესაძლებელი იყოს არა მარტო ღამის საათებში. კარგი სკანირების პროგრამას სჭირდება საკმარისად დაბალი სიჩქარე, იმისათვის, რომ ქსელის სკანირება შესაძლებელი იყოს ნებისმიერ დროს. ოღონდ ნელი კავშირების ან კავშირების დიდი რაოდენობის გარემოში მაინც ჯობია სკანირების გაშვება ღამის საათებში, ქსელზე მინიმმალური გავლენისათვის.

სისტემაზე მინიმალური გავლენა

არ აქვს მნიშვნელობა რა ინსტრუმენტი გამოიყენება უსაფრთხოების შეფასების შესრულებისთვის, სკანირებამ შეიძლება გამოიწვიოს გაუთვალისწინებელი შედეგები სკანირებულ სისტემაში. მაგალითად, ძველი დრაივერით პრინტერები ან როუტერები და გარკვეული ძველი ოპერაციული სისტემები ცუდათ რეაგირებენ სკანირებაზე. ამიტომ სკანირების პროგრამამ უნდა გაითვალისწინოს ეს, სკანირების მინიმალური გავლენისათვის სისტემაზე.

ინტუიციური და კონფიგურირებადი მოხსენების ძრავი

მოწყვლადობის შეფასების მთავარი იდეა არის ინფორმაციის შეგროვება, ამიტომ ძალიან მნიშნელოვანია მოწყვლადობის შეფასების პროგრამული უზრუნველყოფის კარგი მოხსენების გაკეთების შესაძლებლობა. ინსტრუმენტი რომელშიც ყველა წინააღნიშნული თვისებაა შესანიშნავად ახორციელებს ანალიზს, გახდება ნაკლებად ღირებული თუ მას არ აქვს საშუალება გახდოს მიღებულ ინფორმაცია ადვილად გასაგები.

კონფიგურირებადი შემოწმებები

ძალიან მნიშნელოვანია, რომ სკნირების ინსტრუმენტი იყოს ადვილად კონფიგურირებადი და რომ ამ პროგრამის მომხმარებელს ჰქონდეს საშუალება ისე შეასრულოს სკანიბა, როგორც მას უნდა და არა ისე, როგორც მწარმოებელმა მოიფიქრა. იდეალური მოწყვლადობის შეფასების ინსტრუმენტი საშუალებას აძლევს მომხმარებელს დააკონფიგურიროს ტესტები ან საერთოდ შექმნას ახალი შემოწმებების სცენარი სპეციალურად იმ პრობლემის აღმოჩენისთვის, რომელიც ყველაზე მნიშნელოვანია მისთვის.

საწარმოს მასშტაბის გაფართოების შესაძლებლობა

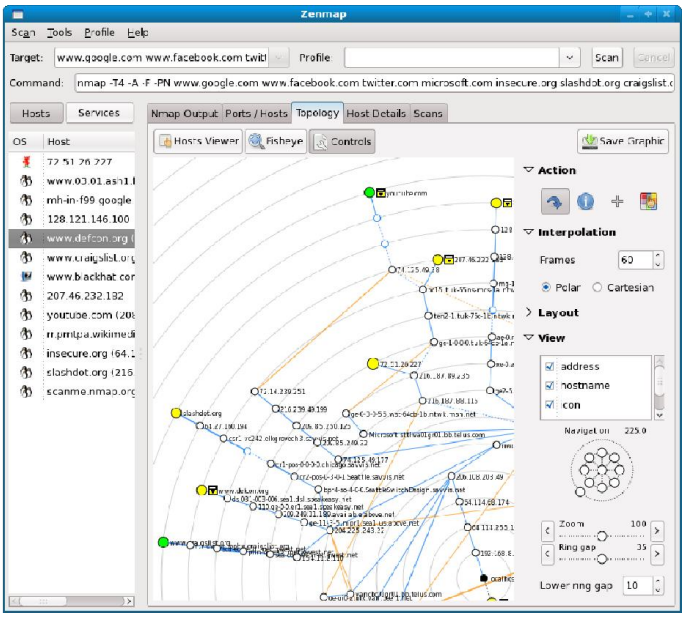
ყველა წინააღნიშნული თვისება სწრაფად ხდებიან უსარგებლო თუ მოწყვლადობის შეფასების ინსტრუმენტი არ გაუძლებს დიდი საწარმოს ქსელებში მუშაობას. ზოგიერთი საუკეთესო ინსტრუმენტები და საუკეთესო იდეები ინსტრუმენტებისთვის კარგავს თავის მნიშნელობას, როდესაც ასეთი ინსტრუმენტები არ ფუნქციონირებენ კარგად ბევრი კომპიუტერების მქონე გარემოში.

იმათვის, რომ იყო გაფართოებადი მოწყვლადობის შეფასების ინსტრუმენტი იყოს ღირებული უნდა გაითვალისწინოს, რომ კორპორაციულ ქსელში სკანირება დააბურუნებს ინფორმაციის დიდი რაოდენობას და უნდა შეძლოს ამ ინფორმაციის დამუშავება. დღეს ბაზარზე არსებული ინსტრუმენტებისათვის ეს საკმაოდ ძნელია, ამითომ თუ უსაფრთხოების შემოწმების პროგრამული უზრუნველყოფა არ იძლევს საშულებას კორპორაციული ქსელებში მუშაობის, მან უნდა დააკმაყოფილოს ყველა ზემოთაღნიშნული თვისებებს მაინც.

ქსელის სკანირების ინსტრუმენტები

ქსელის სკანერები გამოიყენება ავტომატური IP დიაპაზონის სკანირებისთვის იმისათვის, რომ აღმოვაჩინოთ დაუცველი ქსელის მომსახურების კომპონენტები. ორი ყველაზე პოპულარული open source სკანერი არის Nmap და Nessus.

Nmap



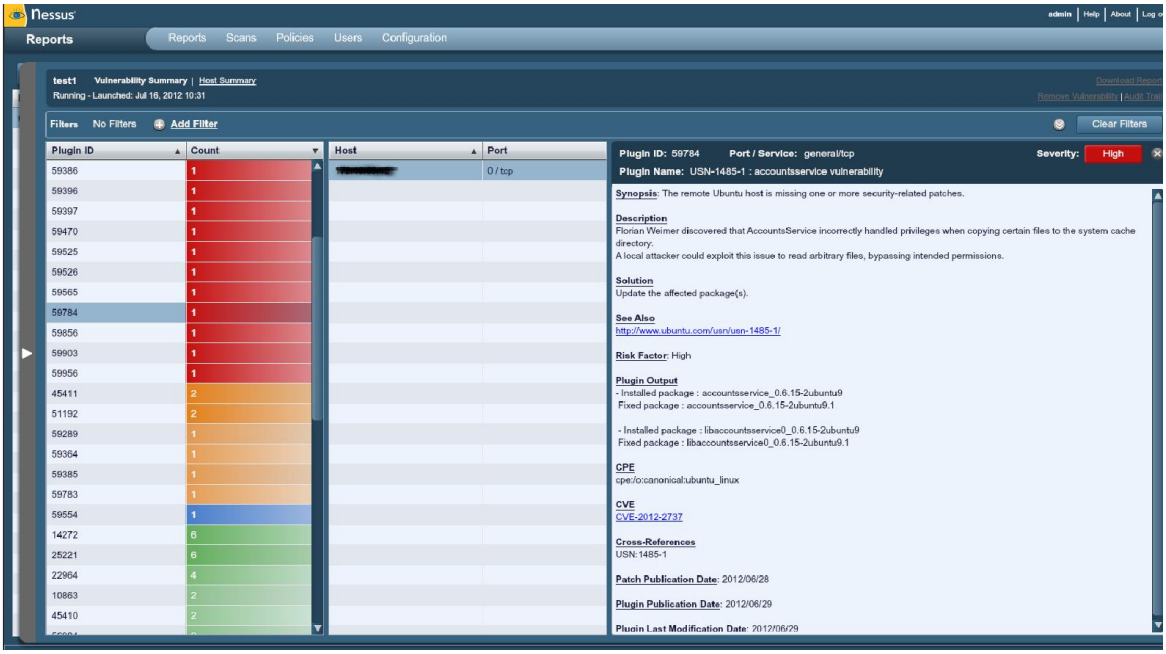
Nmap არის პორტების სკანერი, რომელსაც იყენებენ დიდი ქსელების სკანირებისთვის და დაბალ დონის ICMP, TCP და UDP ანალიზისათვის. Nmap მხარს უჭერს დიდი რაოდენობის სკანირების ტექნიკას და ასევე გვთავაზობს ისეთი მოწინავე თვისებებს, როგორცაა ფინგერპრინტინგი პროტოკოლის სერვისი, IP-ის ფინგერპრინტინგი, დამალული სკანირება და დაბალი დონის ტრაფიკის ფილტრაციის ანალიზი. სისტემის

და ქსელის ბევრი ადმინისტრატორი იყენებს მას ქსელის ინვენტარიზაციისთვის, მომსახურების განახლებების მართვისთვის და ჰოსტის ამ სერვისის მუშაობის მონიტორინგისთვის. Nmap იყენებს ნედლეული IP პაკეტებს ნოვაციური გზებით იმისათვის, რომ განსაზღვროს რა ჰოსტებია ხელმისაწვდომი ქსელში, რა სერვისებს (აპლიკაციის დასახელება და ვერსია) გვთავაზობენ ეს ჰოსტები, რა ოპერაციულ სისტემაში მუშაობენ ისინი, რა ტიპის პაკეტების ფილტრები და firewall-ებია გამოყენებული ყოველ ჰოსტზე და უამრავი სხვა ინფორმაცია. ის იყო შემუშავებული, რათა სწრაფად დაასკანეროს დიდი ქსელები, მაგრამ კარგად მუშაობს ერთი ჰოსტის სისტემაშიც. Nmap მუშაობს ყველა ძირითადი კომპიუტერის ოპერაციული სისტემებში. კლასიკური command-line დამატებით Nmap აქვს მოწინავე ვიზუალური ინტერფეისი, შედეგები მაყურებელს (Zenmap), მოქნილი მონაცემების გადაცემას, გადამისამართების და debugging- ის ინსტრუმენტი (Ncat), სკანირების შედეგების შედარების უტილიტა (Ndiff) და პაკეთი გენერატორი და ანალიზის ინსტრუმენტი (Nping).

Nmap ასევე გვთავაზობს შემდეგ მახასიათებლებს:

- ჰოსტების აღმოჩენა – აიდენტიფიცირებს ქსელის ჰოსტებს. მაგალითად ყველა ჰოსტის ჩამონათვალს, რომელიც პასუხობს TCP და/ან ICMP მოთხოვნაზე ან კონკრეტული პორტი აქვს გახსნილი.
- პორტების სკანირება – სამიზნე ჰოსტზე გახსნილი პორტების აღრიცხვა.
- ვერსიების გამოვლენა – ქსელის სერვისების დაკითხვა იმისათვის, რომ გაარკვიოს დისტანციური მოწყობილობის აპლიკაციის სახელი და ვერსია.
- ოპერაციული სისტემის გამოვლენა – ქსელის მოწყობილობის ოპერაციული სისტემისა და ტექნიკის მახასიათებლების გამოვლენა.
- სამიზნო მოწყობილობასთან ურთიერთქმედების სკრიპტიების დაწერა – ამისათვის არსებოს Nmap Scripting Engine (NSE) და Lua პროგრამირების ენა.
- Nmap შეუძლია შემოგვთავაზოს დამატებითი ინფორმაცია ქსელის მოწყობილობებზე, მაგ. ისეთი როგორიც DNS სახელები, მოწყობილობების ტიპი და MAC მისამართები.

Nessus



Nessus არის მოწყვლადობის შეფასების პაკეტი, რომელსაც შეუძლია შეასრულოს ბევრი ავტომატიზირებული ტესტი სამიზნე ქსელში, მათ შორის ICMP, TCP და UDP სკანირება, სპეციფიური ქსელის მოწყობილობის ტესტირება და მოპოვებული უსაფრთხოების ხარვეზების კარგი მოხსენება. ბევრი ორგანიზაცია იყენებს Nessus ნაყარი ქსელის სკანირებისთვის და უსაფრთხოების შეფასებისთვის. Nessus-ის მოხსენების მიღების შემდეგ იწყებენ უფრო ღრმა ტესტირებას, სპეციალური ტექნიკებით და ინსტრუმენტების გამოყენებით.

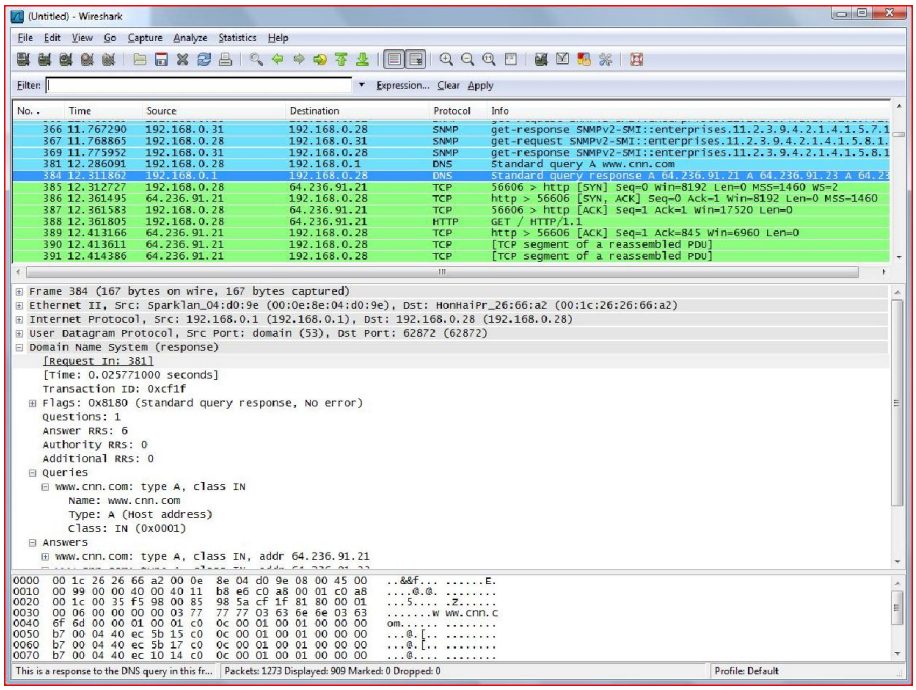
Nessus შედგება ორი კომპორენტისაგან (დაემონი და კლიენტი) და განათავსება ქსელში გადანაწილებული სახით, რაც გვაძლევს ქსელის ტესტებით დაფარვას და ეფექტური ქსელის მართვის საშუალებას.

Nessus სრულყოფილ მოხსენების შედეგებს იძლევა უმრავლეს შემთხვევაში, მაგრამ ეს მოხსენებები ხშირად შეიცავენ რამოდენიმე ცრუ-დადებით შედეგს და ბევრ ხმაურს (ხანდახან ხარვეზები არ არიან სრულად მოხსენებული ან ერთი შეცდომა მოხსენებულია რამდენიმეჯერ), ამიტომ მნიშნელოვანია, რომ უსაფრთხოების კონსულტანტმა ხელით დაფაროს Nessus-ის შედეგი, დააკვალიფიციროს იგი და გააკეთოს ზუსტი და ლაკონური ხელნაწერი ანგარიში. ისევე, როგორც ბევრი სხვა ინსტრუმენტი Nessus იყენებს CVE მითითება შეცდომების მოხსენებისთვის.

CVE არის საერთო ხარვეზების დეტალური ჩამონათვალი წარმოებულია Mitre Corporation მიერ.

Nessus შექმნილია Tenable Security ყველა არსებული პლატფორმისთვის და შესაძლებელია მისი 7 დღიანი უპასო შეფასება.

Wireshark



Wireshark აძლევს საშუალებას ტესტერს ნახოს ქსელიში ყველა მონაცემთა პაკეთი და დაინახოს მთლიანი ქსელის ტრაფიკი. ამ ინსტრუმენტს გააჩნია ბევრი კარგი თვისება, როგორცაა მაგალითად საშუალება მონაცემების დაფიქსირება მუშა ქსელის კავშირის დროს ან ფაილის წაკითხვა რომელშიც ჩაწერილის უკვე დაფიქსირებული ქსელის პაკეტები. Wireshark საშუალება აქვს წაიკითხოს მონაცემები სხვადასხვა Ethernet, IEEE 802.11, PPP, და loopback ქსელებიდან.

დაფიქსირებული ქსელის მონაცემები შეგვიძლია დავათვალიეროთ გრაფიკული ინტერფეისის მეშვეობით, რომელიც ასევე გვაძლევს საშუალებას გავაფართოოთ ის plug-in-ებით. ასევე Wireshark-ს შეუძლია დააფიქსიროს VoIP პაკეტები და ნედლეული USB ტრაფიკს. Wireshark არის ერთ-ერთი ყველაზე მოსახერხებელი ინსტრუმენტი ტესტირებითვის.

კომერციული ქსელის სკანირების ინსტრუმენტები

კომერციული სკანირების ისტრუმენტები გამოიყენება ქსელის ბევრი ადმინისტრატორებისა და დიდი ქსელების უსაფრთხოების პასუხისმგებელი პირების მიერ. ეს კომერციული ინსტრუმენტები, როგორც წესი, არ არიან იაფი (ხშირად ათი ათასობით დოლარის ღირებულების ლიცენზიებით), მაგრამ ასეთი სისტემები შემუშავებულია პატივცემული მწარმოებლების მიერ, რომლებიც მხარს უჭერენ თავისი ინსტრუმენტის განვითარებას და მაგისი დაუცველობის მონაცემთა ბაზების განახლებას. ამ დონის პროფესიული მხარდაჭერით, ქსელის ადმინისთრათორს შეუძლია დაარწმუნოს მისი ქსელის უსაფრთხოების ეფექტურების მაღალ დონეში.

პოპულარული კომერციული ინსტრუმენტები:

ISS Internet Scanner (http://www.iss.net)

eEye Retina (http://www.eeye.com)

QualysGuard (http://www.qualys.com)

Matta Colossus (http://www.trustmatta.com)

ერთ-ერთი პრობლემა ასეთ ავტომატიზირებულ სისტემებში არის დიდი რაოდენობის ცრუ-დადებითი (false positive) შედეგების დაფიქსირება. ამიტომ ხშირად რეკომენდირებულია კომერციული სკანირების ინსტრუმენტების გამოყენება თავდაპირველი ქსელის შეფასებისა და სკანირებისათვის და ამის შემდეგ უკვე ხელით გაწერილი ინსტრუქციებით გაანალიზოთ შესაძლებელი უსაფრთხოების ხარვეზები იმისათვის, რომ უფრო ზუსტი შედეგი იყოს მიღებული. რამდენიმე ინსტრუმენტი სკანირების დასრულების შემდეგ ასევე საბოლოო მოხსენების ხელით რედაქტირების საშუალებას გვაძლევს.

ექსპლუატაციის ფრეიმვორკები (Exploitation Frameworks)

იმის შემდეგ რაც სკანირების მეშვეობით იდენტიფიცირებულია დაუცველი ქსელის სერვისები ექსპლუატაციის ფრეიმვორკების გამოყენებით ამ ქსელის ხარვეზებით ხდება წვდომა სერვისებზე და შეტევა მივიღოთ სამიზნე ჰოსტზე.

ამ ხარვეზების შეფასება მნიშნელოვანია რათა კლიენტს წარმოვუდგინოთ ზუსტი ანგარიში. ამ დროისათვის ერთ-ერთი ყველაზე კარგი და უფასო ფრაიმვორკია Metasploit. ასევე ორი პოპულარული კომერციული ფრაიმვორკერია CORE IMPACT და Immunity CANVAS.

Metasploit ფრეიმვორკი

Metasploit ფრეიმვორკი (MSF) (http://www.metasploit.com) არის მოწინავე ღია პლატფორმა დეველოპმენტისათვის, ტესტირებისთვის და ეკსპლოით კოდის გამოყენებისათვის. პროექტი პირველად შექმნილი იყო როგორც უსაფრთხოების სპეციალისტების შორის პორტატული ქსელის თამაში, მაგრამ შემდგომ განვითარდა ძლიერ ხელსაწყოში პენტესტირებისათვის, ეკსფლოიტების დეველოპმენტისა და დაუცველობის კვლევისათვის.

ფრაიმვორკი და ეკსპლოიტ სკრიპტები დაწერიალია Ruby პროგრამულ ენაზე და ამ ენის ფართო მხარდაჭერა საშუალება გვაძლევს Metasploit-მა იმუშაოს თითქმის ყველა Unix სისტემაში ნაგულისხმევი (default) კონფიგურციით. თვითონ სისტემის კონტროლი შესაძლებელია commandline-ის ინტერპრეტატორის მიერ ან ვებ ინტერფეისით, რომელიც შესაბამისად განთავსებული უნდა იყოს სათანადო სერვერზე.

Metasploit-ის ექსპლოიტ მოდულები საიმედონი არიან და ყველაზე პოპულარული უსაფრთხოების ხარვეზების გამოყენების საშუალებას გვაძლევენ, რომლებიც აღმოჩენილნი იყვნენ Windows და Unix პლატფორმებზე 2004 წლიდან დღემდე. ერთ-ერთი ყველაზე სასარგებლო ფუნქცია მიმდენარე ვერსიაში არის რევერსირებულ VNC სერვერზე ინექციის მექანიზმი (serve injection mechanism), რომელიც ძალიან სასარგებლო არის Windows სერვერების ტესტირების დროს.

კომერციული ექსპლუატაციის ფრეიმვორკები

უსაფრთხოების კონსულტანტები იყენებენ კომერციული ექსპლუატაციის ფრეიმვორკებს იმისათვის, რომ შეასრულონ ქსელებში შეღწევა (penetration) და ქსელის უსაფრთხოების ღრმა შეფასება.

ორი წამყვანი კომერციულად ხელმისაწვდომი ექსპლუატაციის ფრეიმვორკი არის CORE IMPACT და Immunity CANVAS. ამ ინსტრუმენტებს გააჩნიათ ბევრი შესაძლებლობა, ისინი არიან საიმედონი და მათი მწარმოებლები ფინანსურად მხარს უჭერენ მათ განვითარებას. ისინი გვთავაზოვენ მოწინავე თვისებებს, რომლებიც არ გააჩნიათ სხვა ბევრ ფრეიმვორკებს.

დისტრიბუციული კომპანიები (მაგალითად, Argeniss და GLEG) სთავაზობენ zero-day ეკსპლოიტ პაკეტებს, რომლებიც შეიძლება ინტეგრირებული იყოს ზემოთ აღწერილ სისტემებში იმისათვის, რომ შეასრულონ zero-day ტიპის შეტევები.

მიმდინარე ინფორმაცია IMPACT და CANVAS ინტრუმეტების შესახებ ხელმისაწვდომია შემდეგ საიტებზე:

CORE Security Technologies (http://www.coresecurity.com)

Immunity Inc. (http://www.immunityinc.com/products-canvas.shtml).

ასევე ინფორმაცია GLEG და Argeniss 0day პაკეტებზე, რომლებიც შეიცავენ მრავალ გამოუქვეყნებელი ეკსპლოიტ სკრიპტებს გამოქვეყნებულია შემდეგ საიტებზე:

GLEG VulnDisco (http://gleg.net/products.shtml)

Ageniss Ultimate 0day Exploits Pack (http://www.argeniss.com/products.html).

ვებ აპლიკაციების ტესტირების ინსტრუმენტები

ვებ აპლიკაციების ტესტირების ინსტრუმენტები გამოიყენება იმისათვის, რომ შეასრულოთ უსაფრთხოების შეფასება ხელმისაწვდომ ვებ აპლიკაციებსა და კომპონენტებში და მოხდეს მათში სისუსტების აღმოჩენა, ისეთიბის როგორცაა ბრძანების ინექცირება (command injection), XSS (cross-site scripting) და არასწორად მინიჭებული უფლებების გამოყენება.

ასეთი ვებ აპლიკაციის ტესტირების ინსტრუმენტები შეიძლება გავუშვათ ორი გზით, როგორც პასიური პროქსი, რომელიც მონაცემებს ცვლის ბრაუზერიდან იმ დროს როდესაც ის აგზავნის მოთხოვნებს სამიზნე ვებ სერვერზე, ან როგორც აქტიური სკანერი, რომელიც პირდაპირ ამოწმებს შემავალ ცვლადებს. კომპლექსურ ვებ აპლიკაციებში (მაგლითად, ბევრი JavaSript-ის მქონე) ძნელია აქტიური სკანირების გამოყენება, ამიტომ ამ შემთხვევებში პასიური პროქსი უნდა იყოს გამოყენებული.

პროქსიზე დაფუძნებული ღია კოდის ვებ აპლიკაციების ტესტირების ინსტეუმენტები არიან:

Paros (http://www.parosproxy.org)

WebScarab (http://www.owasp.org/index.php/Category:OWASP_WebScarab_Project)

Burp suite (http://portswigger.net)

აქტიური ღია კოდის ვებ აპლიკაციების crawling და fuzzing ინსტრუმენტები:

Active open source web application crawling and fuzzing tools are as follows:

Wapiti (http://wapiti.sourceforge.net)

Nikto (http://www.cirt.net/Nikto2)

ასევე ერთ-ერთი პოპულარული ვებ აპლიკაციების ტესტირების ინსტრუმენტი არის Acunetix-ი.

მას აქვს უფასო და ფასიანი ვერსიები და ძირითადად ამოწმებს აპლიკაციებში SQL Injection-ს და XSS სისუსტეებს. ამ ინსტრუმენტს გააჩნია ძალიან ეფექტური აპლიკაციის crawler-ი, რომელიც შეიცავს კლიენტის სკრიპტების ანალიზის ძრავს (client script analyzer engine). Acunetix აგენერირებს დეტალური რეპორტს უსაფრთხოების შეცდომებსა და ხარვეზებზე. ასევე იგი შეიცავს რამოდენიმე მნიშნელოვან უსაფრთხოების მახასიათებლებს, მაგალითად მოდული, რომელიც ამოწმებს ნელ HTTP Denial of Service და ასევე შეუძლია ანგარიშების გენირაცია ISO 27001 სტანდარტის მიხედვით. ეს მნიშნელოვანია ტესტერებისთვის და დეველოპერებისათვის, რადგან გვაძლევს საშუალებას ორგანიზაციებისთვის შეამოწმოს ამ სტანდარტის მიხედვითაა გაკეთებული ვებ აპლიკაცია თუ არა.

კომერციული ვებ აპლიკაციების სკანირების ინსტრუმენტები

ბევრი კომპანია გთავაზობს ფინანსურად ხელმისაწვდომი ვებ აპლიკაციების ტესტირების ინსტრუმენტებს. მაგალითად სამი ასეთი სკანერი:

Watchfire AppScan (http://www.watchfire.com/products/appscan/)

SPI Dynamics WebInspect (http://www.spidynamics.com/products/webinspect/)

Cenzic Hailstorm (http://www.cenzic.com/products_services/cenzic_hailstorm.php)

სხვა ინსტრუმენტები

w3af - არის ვებ აპლიკაციის შეტევისა და აუდიტის ფრეიმვორკი. ზოგიერთი მახასიათებლებია: სწრაფი HTTP მოთხოვნები, ვებ და პროქსი სერვერების ინტეგრცია კოდში, payload-ების ინექცირება სხვადასხვა სახის HTTP მოთხოვნებში. ამ ინსტრუმენტს გააჩნია ბრძანების ხაზის (command-line) ინტერფეისი და მუშაობს ყველა პოპულარულ პლატფორმაზე Linux, MacOS, Windows. პროგრამის ყველა ვერსია უფასოა.

Aircrack-ng - არის ქსელის უსაფრთხოების ინსტრუმენტების სრულყოფილი კომპლექტი, რომელიც მოიცავს: aircrack-ng (WEP და WPA ლექსიკონიანი შეტევებისთვის), airdecap-ng (დაშიფრული WEP ან WPA ფაილების დეშიფრატორი), airmon-ng (ინსტრუმენტი, რომელსაც გადაყავს ქსელური პლატა მონიტორის რეჟიმში), aireplay-ng (პაკეტების ინჯექტორი (packet injector)), airodump-ng (packet sniffer), airtun-ng (ვირტუალური გვირაბის ინტერფეისები), airolib-ng (ინახავს და მართავს ESSID და პაროლების სიებს),

packetforge-ng (შეუძლია დაშიფრული პაკეტების შექმნა ინექცირებისთვის), airbase-ng (აერთიანებს კლიენტებზე თავდასხმის ტექნიკებს), airdecloak-ng (შლის WEP cloaking), airdriver-ng (უკაბელო ქსელის დრაივერების მართვა), airserv-ng (საშუალებას აძლევს შეღწევადობის ტესტერს, წვდომა მიიღოს სხვა კომპიუტერის wireless პლატაზე).

Airolib-ng ემსგავსება easside-ng იმაში, რომ გაშვების საშუალებას აძლევს მომხმარებელს სკანირების ინსტრუმერნტების დისტანციურ კომპიუტერზე. Easside-ng - იძლევა საშუალებას დაუკავშირდეს დაშვების წერტილს (access point) WEP გასაღების გარეშე, tkiptun-ng - WPA/TKIP შეტევებისათვის და wesside-ng – WEP გასაღებების ავტომატური აღდგენის ინსტრუმენტი.

როგორც უსაფრთხოების ინსტრუმენტების უმრავლესობას Aircrack აქვს გრაფიკული ინტერფეისი - Gerix Wifi Cracker. Gerix არის თავისუფლად ლიცენზირებული უსაფრთხოების ინსტრუმენტი GNU General Public License ქვეშ და შედის ასეთი შეღწევადობის ტესტირების Linux დისტრიბუტივებში, როგორიცაა BackTrack და Backbox. Gerix-ის გრაფიკულ ინტერფეისს გააჩნია რამოდენიმე შეღწევადობის ტესტირების ინსტრუმენტი, რომლებიც ეხმარება ქსელის ანალიზში, wireless პაკეტების დაკავებებში და SQL injection-ში.

Cain & Abel, ან უბრალოდ Cain არის პაროლის აღდგენის ინსტრუმენტი.

ეს ინსტრუმენტი შეღწევადობის ტესტერს ეხმარება სხვადასხვა სახის პაროლების აღდგენაში ქსელის sniffing-ის და დაშიფრული პაროლების გაშიფრაში, ან ლექსიკონით ან brute-force თავდასხმების დროს. Cain-ს ასევე შეუძლია ჩაიწეროს VoIP საუბრები და საშუალება აქვს გაშიფროს დაზიანებული პაროლები, აღმოაჩინო WiFi ქსელის გასაღები და დაქეშირებული პაროლები.

სწორი გამოყენებითა და საჭირო ცოდნით შეღწევადობის ტესტერს ასევე შეუძლია გაანალიზოს მარშრუტიზაციის პროტოკოლები. ეს უსაფრთხოების ინსტრუმენტი არ იყნებს პროგრამული უზრუნველყოფის ხარვეზებს, არამედ აიდენტიფიცირებს უსაფრთხოების სისუსტეებს პროტოკოლის სტანდარტებში ფარგლებში. იგი ასევე გამოიყენება APR (Arp Poison Routing) შეტევების შესწალისთვის. APR აძლევს შესაძლებლობას ქსელის sniffing-ის LAN-ზე და Man-in-the-Middle შეტევების შესრულებას. Sniffer-ის მახასიათებლები პროგრამის ბოლო ვერსიებში საშუალებას იძლევა გაანალიზოთ დაშიფრული პროტოკოლები, ასეთი როგორცაა SSH-1 და HTTPS. ასევე ბოლო ვერსიებში არიან მარშრუტიზაციის პროტოკოლების ავტორიზაციის მონიტორები, ლექსიკონი და უხეში ძალის კრეკერი ყველა პოპულარული hash

ალგორითმებისთვის, პაროლების კალკულატორები, კრიპტოანალიზის თავდასხმები და პაროლის დეკოდერების ინსტრუმენტები.

Ettercap არის თავისუფალი და ღია ქსელის უსაფრთხოების ინსტრუმენტი man-in-the-middle (MITM) შეტევიბის შესრულებისათვის LAN-ში.
ეს უსაფრთხოების ინსტრუმენტი შეიძლება გამოყენებული იყოს კომპიუტერული ქსელის პროტოკოლების ანალიზში, უსაფრთხოების აუდიტის კონტექსტის ფარგლებში.
Ettercap გააჩნია ფუნქციონირების ოთხი მეთოდი:

- უსაფრთხოების სკანირება IP პაკეტების ფილტრაციით;
- MAC-ზე დაფუძნებული, როდესაც პაკეტები იფილტრება MAC-ის საფუძველზე (gateway sniffing);
- ARP-ზე დაფუძვნებული სკანირება ARP poisoning გამოყენენებით იმისათვის, რომ მოუსმინოთ ორ ჰოსტის შორის კავშირს LAN-ზე (full-duplex);
- PublicARP ფუნქციონალი: Ettercap იყენებს ARP poisoning-ს იმისათვის, რომ LAN-ზე მოუსმინოს კავშირს მსხვერპლის ჰოსტიდან ყველა სხვა ჰოსტზე (half-duplex).

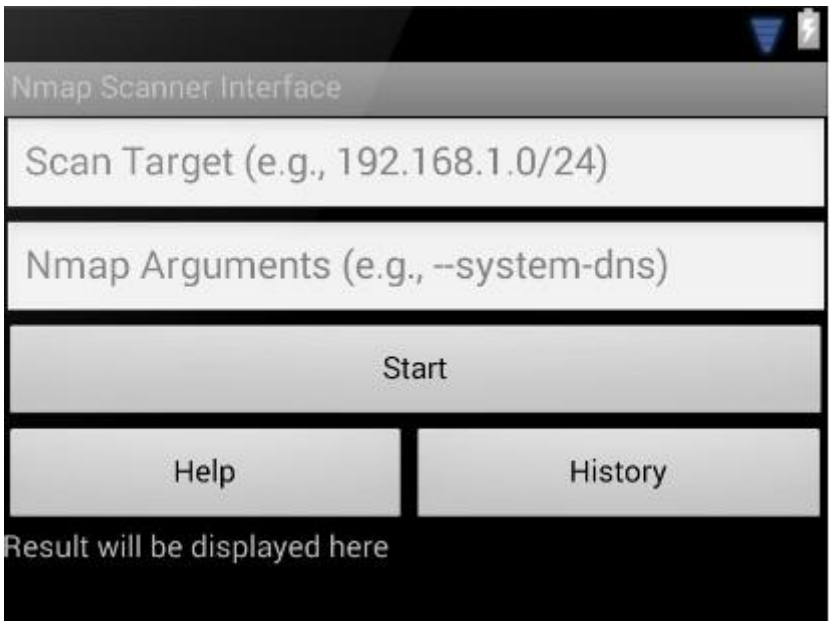
John The Ripper არის ძალიან პოპულრული უსაფრთხოების ინსტრუმენტი, რომელსაც ხშირად ეძახიან უბრალოდ "John" და ის წარმოადგენს პაროლების უფასო გატეხვის პროგრამული უზრუნველყოფას. თავდაპირველად ის შექმნილი იყო Unix ოპერაციული სისტემებისთვის, მაგრამ ამ დროისათვის მუშაობს ყველა ძირითად ოპერაციული სისტემაზე. John არის ერთ-ერთი ყველაზე პოპულარული პაროლების ტესტირებისა და გატეხვის პროგრამა, რომელიც გამოყენებულია ინფორმაციის უსაფრთხოების ექსპერტების მიერ. ეს შეხწევადობის ინსტრუმენტი აერთიანებს სხვადასხვა პაროლებების კრეკერებს ერთ ლაკონურ პაკეტში, რომელსაც შეუძლია დააიდენტიფიციროს პაროლის ჰეშის ტიპი თავისი საკუთარი გატეხვის ალგორითმის მეშვეობით. John არამარტო ხსნის დაჰეშირებულ პაროლებს Windows-ზე, არამედ მას დამატებითი პარამეტრების გარეშე შეუძლია გატეხოს ნებისმიერი პაროლი, რომელშიც გამოყენებულია შიფროტექსტები ან ჰეშირების ფორმატები DES, MD5, Blowfish ან AFS.

Kismet არის უკაბელო ქსელის დეტექტორი, სნიფერი და შეტევების გზების აღმოჩენის შეღწევადობის ინსტრუმენტი. Kismet-ს შეუძლია მონიტორინგი გაუწიოს და მოუსმინოს 802.11b, 802.11a, 802.11g, და 802.11n ქსელის ტრაფიკში. არსებობს ბევრი sniffing

ინსტრუმენტი, მაგრამ Kismet პოპულარული გახდა, რადგან მუშაობს პასიურ რეჟიმში, რაც ნიშნავს, რომ ის არ აგზავნის არანაირ ლოგების პაკეტებს wireless დაშვების წერტილსა და უკაბელო კლიენტებზე, როდესაც უკავშირდება მათ. Kismet ღია კოდიანი და ფართოდ გამოყენებადია.

მობილური აპლიკაცია ქსელის შეფასებისათვის

ამ ნაშრომში წარმოდგენილი პროგრამული უზრუნველყოფა არის მცდელობა უფრო მოსახერხებელი ინტერფეისის შექმნა ქსელის ანალიზის ინსტრუმენტისათვის.
ეს ინტერფეისი დაფუძნებულია კონსოლურ Nmap ქსელის ანალიზის ინსტრუმენტზე, რომელიც არის ერთ-ერთი ყველაზე პოპულარული პროგრამა ქსელის შეფასებისათვის.
სფეციალურად Android სისტემისათვის, Nmap-ის მოხმარებლების და შექმნელების მიერ, შექმნილი იყო Nmap-ის ვერსია, რომელსაც საშუალება აქვს იმუშაოს ARM პროცესორებზე, რომელებიც ყველაზე გავრცელებული არიან Android-ის სისტემის მქონე სმარტფონების შორის. მაგრამ პროგრამის დაყენება მაინც რთულია და თვითონ არის მთლიანად კონსოლური, რაც ართულებს მასთან მუშაობას, განსაკუთრებით პატარა ეკრანის მქონე მოწყობილობებზე, სადაც ვირტუალური კლავიატურა იკავებს ეკრანის დიდ ნაწილს.
ამიტომ ამ პროგრამისათვის საჭიროა მოსახერხებელი ინტერფეისი.



ინტერფეისის მთავარ ეკრანზე არის ორი ველი, პირველი სამიზნე IP მისამართისათვის და მეორე Nmap-ის სკანირების პარამეტრებისთვის. მათ ქვეშ არის საკონტროლო

ღილაკები, რომელთა მეშვეობითაც შეგვიძლია დავიწყოთ სკანირება, ვნახოთ Nmap-ის პარამეტრები და წინა განხორციელებული მოთხოვნები. ამ ღილაკების მერე ეწერება სკანირების შედეგები განხორხიელების დრო, კავშირის ხარისხი, გახსნილი პორტი და ა.შ.

Nmap-ს შეუძლია სხვადასხვა მეთოდებით ქსელის სკანირება და შესაბამის ველში მომხმარებელს შეუძლია ჩაწეროს სკანირების პარამეტრები. მოცემული პარამეტრების საფუძველზე მიღებული პასუხი აბრუნებს სხვადასხვა ინფორმაციას დასკანირებული მოწყიბილობის შესახებ.

სკანირების მერე მიღებული ინფორმაცია მომხმარებელს ეხმარება ქსელის სწრაფ ანალიზში. უსაფრთხოების სპეციალისტი გაიგებს რა სისტემაა დაყენებული დასკანირებულ IP-ს მქონე მოწყობილობაზე და რა პორტები არის გახსნილი მასზე, რაც შემდგომში დაგეხმარებათ სატესტო შეტევის განხორციელებაში ან უბრალოდ გამოავლენს უსაფრთხოების პრობლემას.

დასკვნა

ბოლო წლების განმავლობაში ინტერნეტის სწრაფი განვითარების პირობებში, საჭიროა უსაფრთხოების ხარვეზების ძებნის ახალი მეთოდებისა და ინსტრუმენტების განვითარება, როგორც დიდი კორპორაციული, ასევე პატარა ლოკალური ქსელებისთვის. მობილური ტექნოლოგიების განვითარებასთან ერთად უსაფრთხოების სპეციალისტებს ხანდახან სჭირდებათ ქსელის უსაფრთხოების სწრაფი ანალიზი, მასთან ხელმისაწვდომი ტექნიკით, რომლისათვისაც შეიძლება გამოვიყენოთ სმარტფონები ან პლანშეტები.

ძირითადად ქსელის უსაფრთხოების ტესტერმა იმისათვის, რომ დაასკანეროს და გაანალიზოს ქსელი უნდა გამოიყენოს ნოუთბუქი ან დესქტოპ კომპიუტერი, მობილური პლატფორმებზე ისეთი ინსტრუმენტების რაოდენობა არის მცირე. უსაფრთხოების სპეციალისტებს დასჭირდებათ მობილური პლატფორმისთვის დაწერილი უსაფრთხოების ინსტრუმენტები, რადგან ტექნოლოგიების განვითარება ტრადიციული დესქტოპ კომპიუტერიდან უფრო მეტად გადადის მობილურ პლატფორმებზე.

გამოყენებული ლიტერატურა

1. CCNA Study Guide (http://computernetworkingnotes.com/cisco/ccna-study-guide/)
2. Network Security Assessment 2nd Edition, Chris McNab, O'Reilly Media 2007
3. The Open Web Application Security Project (OWASP)
4. Lye K., Wing J. Game Strategies in Network Security, International Journal of Information Security.
5. An Introduction to Computer Security: The NIST Handbook, Barbara Guttman and Edward Roback
6. The Hacker Always Gets Through, T.J. O'Connor, 2014
7. White-Hat Security Arsenal, Aviel Rubin, 2001
8. Nmap documentation (http://insecure.org)
9. Android Developer's Guide (https://developer.android.com/guide/index.html)