ვირტუალურ კერძო ქსელებში (VPN) სიმბოლოების დაშიფვრის კომბინირებული მეთოდი

ოთარ შონია, იოსებ ქართველიშვილი, ლუკა შონია, ზებურ ბერიძე
საქართველოს ტექნიკური უნივერსიტეტი

რეზიუმე

წარმოდგენილია ვირტუალურ კერძო ქსელებში სიმბოლოების დაშიფვრის კომბინირებული მეთოდი. დეტალურად აღწერილია მეთოდის თითოეული ეტაპი. სისტემაში გამოყენებული სპეციალური პარამეტრების წყალობით დაშიფვრისა და ამოშიფვრის გასაღები არის უნიკალური და სისტემაში მომხმარებლის ყოველი ავტორიზაციის დროს გასაღებები იცვლება და არასდროს არ განმეორდება.

საკვანძო სიტყვები: ვირტუალური ქსელები. სიმბოლოების დაშიფვრა. ამოშიფვრის გასაღები. კრიპტოგრაფიული მეთოდი. მონაცემების უსაფრთხოება.

1. შესავალი

როგორც მოგეხსენებათ, ორ მოცილებულ კომპიუტერს შორის, რომელიც იყენებს გლობალური ქსელის – ინტერნეტის ინფრასტრუქტურას, უსაფრთხო კავშირის არხის შექმნისთვის, ვირტუალური კერძო ქსელების VPN (Virtual Private Network) აგების ტექნოლოგია დღეისათვის წარმოადგენს ერთ-ერთ ყველაზე ოპტიმალურ ვარიანტს. წამოჭრილი ამოცანა ძალზედ მნიშვნელოვანია, ვინაიდან საიმედო კავშირი, სადაც შესაძებელია გადაიცეს კონფიდენციალური ინფორმაცია, უბრალოდ აუცილებელია ადამიანის მოღვაწეობის უამრავ სფეროში, მაგალითად, საბანკო საქმეში, ელექტრონულ კომერციაში და სხვა. აქედან გამომდინარე, ვირტუალური კერძო ქსელები ძალზედ მოსახერხებელია აღნიშნული ამოცანის გადასაჭრელლად და ადამიანების უმრავლესობა გლობალურ ქსელში სხვადასხვა კავშირების დასამყარებლად VPN ტექნოლოგიას მიიჩნევს ერთ-ერთ ყველაზე მძლავრ და მოსახერხებელ საშუალებად. თუმცა, საქმე მთლად ასე მარტივად არ წარმოგვიდგება. ვირტუალურ კერძო ქსელებს გააჩნია თავიანთი ნაკლოვანებები და სუსტი მხარეები. ვირტუალური კერძო ქსელების ტექნოლოგია აგებულია კრიპტოგრაფიული მეთოდების გამოყენებაზე. კერძოდ, ყველა ინფორმაცია, რომელიც მიედინება დაცული კავშირის არხში, იმყოფება დაშიფრულ მდგომარეობაში. აქედან გამომდინარე, VPN-ის საფუძველს წარმოადგენს კრიპტოგრაფია და მის არეალში აგრეთვე მოქმედებს ზოგიერთი დამატებითი მექანიზმები, როგორებიცაა, მაგალითად მომხმარებლების აუტენტიფიკაცია, მონაცემთა მთლიანობის კონტროლი და სხვა. თუმცა, კრიპტოგრაფიულ მეთოდებს გააჩნიათ თავიანთი სუსტი ადგილები.

ნებისმიერი კრიპტოგრაფიული მეთოდის გამოყენების საიმედოობა დაფუძნებულია მასში გამოყენებულ დაშიფვრის ალგორითმზე. რა თქმა უნდა, მონაცემების სუსტი დაშიფვრა ბოროტგანმზრახველს საშუალებას აძლევს ადვილად მოიპოვოს წვდომა მისთვის სასურველ ინფორმაციაზე. ბუნებრივია ჩნდება კითხვა, თუ რომელი კრიპტოგრაფიული მეთოდი უნდა იქნას გამოყენებული მონაცემების უსაფრთხოებისთვის.

დღეისათვის გამოიყენება ღია და დახურული კრიპტოგრაფიული ალგორითმები. ღია ალგორითმების ჯგუფს მიეკუთვნება ისეთი ცნობილი ტექნოლოგიები, როგორებიცაა: DES (Data Encryption Standard), TripleDES (Triple Data Encryption Algorithm), RSA (Rivest, Shamir and Adleman), AES (Advanced Encryption Standard) და სხვა. ისინი გაერთიანებულია სხვადასხვა ქვეყნის ნაციონალურ სტანდარტებში. დახურული კრიპტოგრაფიული ალგორითმები მუშავდება სხვადასხვა კომპანიების მიერ და გამოიყენება თავიანთ საკუთრებაში. ინფორმაციის დაშიფვრისთვის გამოიყენება კრიპტოგრაფიული გასაღები და დიდი მნიშვნელობა ენიჭება

დაშიფვრის მექანიზმს და გასაღების სიგრძეს. ვინაიდან, რაც უფრო რთულია დაშიფვრის მექანიზმი და დიდია გასაღების სიგრძე, მით უფრო გაუჭირდება ბოროტგანმზრახველს მისი ამოცნობა.

ყველა ზემოთგანხილული არსებული თუ აქამდე შემოთავაზებული კრიპტოგრაფიული ტექნოლოგიები დაფუძნებულია მატრიცული მეთოდების გამოყენებაზე. რა თქმა უნდა, რაც უფრო გართულებულია მონაცემების დაშიფვრის მექანიზმი, მით უფრო გაუჭირდება ბოროტგანმზრახველს მისი გაშიფვრა. თუმცა ამასთან ჩნდება ახალი პორბლემა, ეს არის მონაცემების გადაცემის სიჩქარე. დაშიფვრის მექანიზმის ძალიან გართულებას მოსდევს მონაცემების გადაცემის სიჩქარის შემცირება, მითუმეტეს მაშინ, როდესაც მომხმარებელს უწევს მუშაობა განაწილებულ მონაცემთა ბაზების მართვის სისტემასთან. მონაცემთა ბაზაში ჩანაწერების რაოდენობის გაზრდა იწვევს მომხმარებლის მიერ მოცილებული სამუშაო ადგილიდან მონაცემთა წამოღების სიჩქარის ვარდნას. მონაცემთა ბაზების ოპტიმიზაციის პრობლემა დღეისათვის წარმოადგენს ერთ-ერთ საპრობლემო სფეროს და დღეისათვის აქტიურად მიმდინარეობს მუშაობა ამ პრობლემის აღმოსაფხვრელად. აქედან გამომდინარე უნდა შემუშავდეს ისეთი დაშიფვრის მექანიზმი, რომელიც თავისი თვისებებით იქნება მარტივი, დიდ გამოთვლით პროცესებთან არ იქნება დაკავშირებული და, რაღა თქმა უნდა, გარეშე უცხო პირისთვის მისი ამოცნობის ალბათობა იქნება ძალზედ მცირე. არსებულ დაშიფვრის მეთოდებს გააჩნია კიდევ ერთი პრობლემა. დაშიფვრის გასაღები უმრავლეს შემთხვევაში არ არის ცვალებადი, ან არის იშვიათად ცვალებადი, რაც ბოროტგანმზრახველს ხელს უწყობს გარკვეული დროის განმავლობაში მოახდინოს მისი გაშიფვრა. ამიტომ, სასურველია სისტემაში შემოღებული იქნას დამატებითი პარამეტრები (კოეფიციენტები), რომელიც გასაღებს გახდის ცვალებადს. კერძოდ, სისტემაში მომხმარებლის ყოველი ავტორიზაციის დროს დაშიფრვის და ამოშიფვრის გასაღები იქნება უნიკალური (ანუ არასდროს განმეორდება) და შეუძლებელი იქნება მისი გატეხვა.

2. ძირითადი ნაწილი

ზემოაღნიშნული პრობლემებიდან გამომდინარე შემუშავებულია სიმბოლოების დაშიფვრის კომბინირებული მეთოდი. სიმბოლოების დაშიფვრა და მისი ამოცნობა, თავისი თვისებებიდან გამომდინარე, შეიცავს განსაკუთრებულ პრობლემებს, რომელთა გადაწყვეტაც წარმოადგენს უსაფრთხოების ავტომატიზებული სისტემის აგების აუცილებელ პირობას. სიმბოლოების დაშიფვრის კომბინირებული მეთოდი მოიცავს შემდეგ ეტაპებს:

1) მომხმარებლის მიერ შეტანილი პაროლის დაშლა სიმბოლოებად;

2) თითოეული სიმბოლოს გადაყვანა ASCII(decimal) კოდირებაში, მათი კოდების განსაზღვრა;

3) მიღებული კოდებით სპეციალური ოპერაციის დახმარებით დამატებითი სიმბოლოების განსაზღვრა სისტემაში დაყენებული პარამეტრის მიხედვით;

4) სიმბოლოების და დამატებითი სიმბოლოების გაერთიანება და მათი სიტყვებად დაშლა (ჯგუფების შექმნა) სისტემაში დაყენებული პარამეტრის მიხედვით;

5) თითოეული ჯგუფის სიმბოლოებისთვის ASCII(decimal) კოდირებაში ხდება მათი შესაბამისი კოდების განსაზღვრა;

6) სპეციალური ოპერაციის დახმარებით მიღებული კოდების რიცხობრივი მნიშვნელობა გარდაიქმნება სხვა რიცხობრივ მნიშვნელობად და მიიღება ახალი კოდების სიმრავლე;

7) მიღებული კოდების სიმრავლისაგან მიიღება სპეციალური სიმბოლოებისაგან შემდგარი ჯგუფები;

8) მიღებული ჯგუფების გაერთიანებით მიიღება სპეციალური სიმბოლოების ერთობლიობისგან შემდგარი დაშიფრული ინფორმაცია.

სიმბოლოების დაშიფვრის კომბინირებული მეთოდის ეტაპები თავისი ფუნქციონალური დანიშნულებებით შეიძლება დახასიათდეს შემდეგნაირად:

პირველ ეტაპზე მომხმარებლის მიერ შეტანილი პაროლი იწერება სპეციალურ მასივში, სადაც განსაზღვრულია სიტყვის დასაწყისი და დასასრული. აგრეთვე ხდება სიტყვის დაშლა სიმბოლოებად და ცალკეული სიმბოლოსთვის განსაზღვრულია მისი ინდექსი.

მე-2 ეტაპზე თითოეული სიმბოლო განიხილება ASCII(decimal) კოდირებაში და მათთვის განისაზღვრება კოდები, რომლებიც იმახსოვრება სპეციალურ ცვლადებში;

მე-3 ეტაპზე ინფორმაციის უსაფრთხოება ბევრად არის დამოკიდებული მომხმარებლებზე. მაგალითად, მომხმარებელმა შეიძლება აირჩიოს ძალიან ადვილი პაროლი, რომელიც ამოსაცნობად მარტივი იქნება უცხო პირისთვის. აქედან გამომდინარე აუცილებელია ავტომატიზებულად მოხდეს მისი ,,გართულება". მეორე ეტაპზე მიღებული კოდებით სპეციალური ოპერაციის დახმარებით სისტემაში დაყენებული პარამეტრის მიხედვით განისაზღვრება მოდიფიცირებული კოდები, რომლებისგანაც მიიღება დამატებითი სიმბოლოები.

მე-4 ეტაპზე მიღებული სიმბოლოებითა და დამატებითი სიმბოლოების გაერთიანებით იქმნება ,,გართულებული" პაროლი, რომელსაც ემატება მიმდინარე თარიღისა და დროის რიცხობრივი მნიშვნელობები. შემდეგ ხდება მისი სიტყვებად დაშლა (ჯგუფების შექმნა) სისტემაში დაყენებული პარამეტრის მიხედვით. სასურველია რომ მომხმარებლის ავტორიზაციის პროცესში ცენტრალური სერვერისთვის პაროლის გადაცემა მოხდეს ნაწილ-ნაწილ (ცალკეულ ჯგუფებად) შუალედური დასტურების ვითარებაში.

მე-5 და მე-6 ეტაპებზე თითოეული ჯგუფის სიმბოლოებისთვის ASCII(decimal) კოდირებაში ხდება მათი შესაბამისი კოდების განსაზღვრა. შემდეგ სპეციალური ოპერაციის დახმარებით მიღებული კოდების რიცხობრივი მნიშვნელობა გარდაიქმნება სხვა რიცხობრივ მნიშვნელობად და მიიღება ახალი კოდების სიმრავლე;

მე-7 და მე-8 ეტაპებზე მიღებული კოდების სიმრავლისაგან მიიღება სპეციალური სიმბოლოებისაგან შემდგარი ჯგუფები და მიღებული ჯგუფების გაერთიანებით მიიღება სპეციალური სიმბოლოების ერთობლიობისგან შემდგარი დაშიფრული ინფორმაცია, რომლის ამოშიფვრაც ცენტრალური სერვერის მიერ ხდება უკუალგორითმის საშუალებით.

განვიხილოთ სიმბოლოების დაშიფვრის კომბინირებული მეთოდის ფორმალური ნაწილი. ამისათვის შემოვიტანოთ აღნიშვნები: $i-$ სიმბოლოს ინდექსი; $j-$ დამატებითი სიმბოლოს ინდექსი; n_1- სიმბოლოების რაოდენობა მომხმარებლის მიერ შეტანილ სიტყვაში (პაროლში), n_2- სიმბოლოების რაოდენობა დამატებით სიტყვაში.

დასაწყისისთვის: $i=1;\quad j=1;\quad n_1=0;\quad n_2=0.$

ვადგენთ მასივს: $S_{\{i\}}F, L \qquad i=\overline{1,n_1}$,

სადაც $S_{\{i\}}F$ არის სიმბოლოების სიტყვის დასაწყისი, $S_{\{i\}}L$ არის სიმბოლოების სიტყვის დასასრული, n_1 არის სიმბოლოების რაოდენობა სიტყვაში.

შემდეგ ხდება სიტყვაში სიმბოლოების დაშლა და ცალკეული სიმბოლოსთვის ინდექსის განსაზღვრა:

$$i=1$$
$$S_{\{i\}}=S_{\{i\}}F$$
$$S_{\{i\}}F=S_{\{i\}}F+1$$
$$i=i+1;\qquad n_1=n_1+1$$

ვიდრე $i \le S_{\{i\}}L$

რის შედეგაც მიიღება სიმბოლოების ერთობლიობისაგან შემდგარი სიტყვა, რაშიც ცალკეული სიმბოლოსთვის განსაზღვრულია თავისი ინდექსი:

$$S_{\{i\}}F, L = (S_{\{1\}}, S_{\{2\}}, \ldots, S_{\{n_1\}})$$

თითოეული სიმბოლო განიხილება ASCII(decimal) კოდირებაში და მათთვის განისაზღვრება კოდები. ამისათვის შემოვიღოთ მასივი:

$$S_{ASCII\{i\}}F, L \qquad i = \overline{1, n_1}$$

სადაც $S_{ASCII\{i\}}F$ არის სიმბოლოების კოდების დასაწყისი, $S_{ASCII\{i\}}L$ არის სიმბოლოების კოდების დასასრული, n_1 არის სიმბოლოების კოდების რაოდენობა.

შემდეგ ხდება ცალკეული სიმბოლოსთვის კოდის განსაზღვრა და მისი ჩაწერა სიმბოლოების კოდების მასივში:

$$i = 1$$
$$S_{ASCII\{i\}} = S_{\{i\}}F, L$$
$$i = i + 1;$$

ვიდრე $i \le n_1$

რის შედეგაც მიიღება სიმბოლოების კოდების ერთობლიობისაგან შემდგარი რიცხვები, რაშიც ცალკეული სიმბოლოსთვის განსაზღვრულია თავისი ინდექსის კოდი:

$$S_{ASCII\{i\}}F, L = (S_{ASCII\{1\}}, S_{ASCII\{2\}}, \ldots, S_{ASCII\{n_1\}})$$

შემდეგ უნდა მოვახდინოთ სიტყვის შევსება დამატებითი სიმბოლოებით. ამისათვის შემოვიტანოთ აღნიშვნა – P_1, რომელიც წარმოადგენს სიტყვის შევსების პარამეტრს, ანუ რა რაოდენობისაგან უნდა შედგებოდეს სიმბოლოებითა და დამატებითი სიმბოლოების გაერთიანებით შედგენილი სიტყვა.

თავდაპირველად უნდა განისაზღვროს დამატებითი სიმბოლოების რაოდენობა – n_2.

$$n_2 = p_1 - n_1$$

ვადგენთ მასივს, სადაც უნდა ჩაიწეროს დამატებითი სიმბოლოები:

$$D_{\{j\}}F, L \qquad j = \overline{1, n_2}$$

სადაც $D_{\{j\}}F$ არის დამატებითი სიმბოლოების სიტყვის დასაწყისი, $D_{\{j\}}L$ არის დამატებითი სიმბოლოების სიტყვის დასასრული, n_2 არის დამატებითი სიმბოლოების რაოდენობა სიტყვაში.

შემდგომ ცალკეული სიმბოლოს განსაზღვრული კოდისთვის განისაზღვრება მოდიფიცირებული კოდები. ამისათვის შემოვიღოთ აღნიშვნა – b_k, რომელიც არის დამატებითი სიმბოლოების შევსებისთვის საჭირო ძირითად სიმბოლოებში გასავლელი ბიჯების რაოდენობა. $k = \overline{1, m}$, სადაც m - არის ბიჯების რაოდენობა.

ამისათვის შემოვიღოთ მასივი, სადაც მოხდება მოდიფიცირებული კოდების ჩაწერა:

$$D_{ASCII\{j\}}F, L \qquad j = \overline{1, n_2} \tag{10}$$

სადაც $D_{ASCII\{j\}}F$ არის დამატებითი სიმბოლოების კოდების დასაწყისი, $D_{ASCII\{j\}}L$ არის დამატებითი სიმბოლოების კოდების დასასრული, n_2 არის დამატებითი სიმბოლოების კოდების რაოდენობა.

მოცემული მეთოდის მიზანი მოდიფიცირებული კოდებით დასაშიფრი სიმბოლოების გადაყვანა სპეციალურ სიმბოლოებში. ASCII(decimal) კოდირების სისტემაში სიმბოლოების კოდები შეესაბამება 32-დან 126-ის ჩათვლით, ხოლო სპეციალური სიმბოლოების კოდები შეესაბამება 128-დან 255-ის ჩათვლით.

სიმბოლოების კოდების გარდაქმნა ხდება შემდეგნაირად. აიღება სიტყვის პირველი სიმბოლოს კოდი $S_{ASCII\{1\}}$, გადამოწმდება მისი მდებარეობის არეალი. თუ კოდი მდებარეობს 32-დან 50-ის ჩათვლით, მაშინ შესრულდება შემდეგი მოქმედება: სიმბოლოს კოდს დაემატება მისი რიგითობა, ანუ ინდექსი, შემდეგ მეორე სიმბოლოს კოდს დაემატება მეორე ინდექსი და ა.შ. სიტყვის ბოლომდე. თუ კოდი მდებარეობს 51-დან 126-ის ჩათვლით, მაშინ აკლდება. მიღებული ახალი კოდებით დგება დამატებითი სიმბოლოების ჯგუფი. ეს პროცესი მიმდინარეობს იქამდე, ვიდრე არ დაკმაყოფილდება p_1 პარამეტრის მნიშვნელობა. თუ b_k-სთვის $k=1$, ანუ პირველი ბიჯია, მაშინ სიმბოლოების რიგითობა იწყებს 1-დან, თუ $k=2$ – მე-2 ბიჯი, 2-დან და ა.შ.

$$b_k = 1$$

თუ $32 \le S_{ASCII\{i\}}F, L \le 50$, მაშინ

$$D_{ASCII\{j\}}F, L = S_{ASCII\{i\}}F, L + S_{ASCII\{i\}}$$

თუ $51 \le S_{ASCII\{i\}}F, L \le 126$, მაშინ

$$D_{ASCII\{j\}}F, L = S_{ASCII\{i\}}F, L - S_{ASCII\{i\}}$$

$$k = k + 1;\ \ i = i + 1;\ \ j = j + 1$$

მიღებული ახალი მოდიფიცირებული კოდებით დგება დამატებითი სიმბოლოების ჯგუფი:

$$j = 1$$

$$D_{\{j\}} = D_{ASCII\{j\}}F, L$$

$$j = j + 1$$

ვიდრე $j \le n_2$

რის შედეგაც მიიღება დამატებითი სიმბოლოების ერთობლიობისაგან შემდგარი სიტყვა, რაშიც ცალკეული სიმბოლოსთვის განსაზღვრულია თავისი ინდექსი:

$$D_{\{j\}}F, L = (D_{\{1\}}, D_{\{2\}}, ..., D_{\{n_2\}})$$

მიღებული სიმბოლოებითა და დამატებითი სიმბოლოების გაერთიანებით იქმნება „გართულებული“ სიტყვა (პაროლი), რომელსაც ემატება მიმდინარე თარიღისა და დროის რიცხობრივი მნიშვნელობები. შემოვიტანოთ აღნიშვნები – *DateTime*, რომელშიც ფიქსირდება მიმდინარე თარიღისა და დროის რიცხობრივი მონაცემი - *DateTime=now()*; $SD_{\{i\cup j\}}F, L$, რაშიც იწერება გაერთიანებული სიტყვა:

$$SD_{\{i\cup j\}}F, L = S_{\{i\}}F, L + D_{\{j\}}F, L + DateTime$$

სადაც $i = \overline{1, n_1};\quad j = \overline{1, n_2}$

შედეგად მიიღება სიმბოლოებისაგან და დამატებითი სიმბოლოების ერთობლიობისაგან შემდგარი სიტყვა, რაშიც ცალკეული სიმბოლოსთვის განსაზღვრულია თავისი ინდექსი:

$$SD_{\{i\cup j\}}F,L=(S_{\{1\}},S_{\{2\}},...,S_{\{n_1\}},...,D_{\{1\}},D_{\{2\}},...,D_{\{n_2\}})$$

შემდეგ ხდება მიღებული სიტყვის დაშლა (ჯგუფების შექმნა) სისტემაში დაყენებული პარამეტრის მიხედვით. ამისათვის შემოვიტანოთ აღნიშვნები – p_2, რომელიც წარმოადგენს დასაშლელი სიტყვის რაოდენობა, ანუ რა რაოდენობით უნდა დაიშალოს მიღებული სიტყვა; $G_{\{\ell\}}SD_{\{i\cup j\}}F,L$, სადაც ფიქსირდება ჯგუფში შემავალი სიმბოლოები და ჯგუფის ინდექსი; m – ჯგუფების რაოდენობა.

დაშლილი სიტყვების რაოდენობა აღვნიშნოთ ℓ-ით და იგი გამოითვლება შემდეგნაირად:

$$\ell=(n_1+n_2+DateTime_{Count})divp_2;$$
$$\ell=\overline{1,m}$$

სადაც div – ნიშნავს გაყოფას ნაშთის გარეშე. $DateTime_{Count}$ – თარიღისა და დროის რიცხობრივ მაჩვენებლებში სიმბოლოების რაოდენობა. შედეგად მივიღებთ ჯგუფების ერთობლიობას:

$$G_{\{\ell\}}SD_{\{i\cup j\}}F,L=(G_{\{1\}}SD_{\{i\cup j\}},G_{\{2\}}SD_{\{i\cup j\}},...,G_{\{m\}}SD_{\{i\cup j\}}) \quad (17)$$

შემდეგ თითოეული ჯგუფის სიმბოლოებისთვის ASCII(decimal) კოდირებაში ხდება მათი შესაბამისი კოდების განსაზღვრა.

$$G_{ASCII\{\ell\}}SD_{\{i\cup j\}}=G_{\{\ell\}}SD_{\{i\cup j\}}F,L$$
$$\ell=\ell+1$$
ვიდრე $\ell\le m$

რის შედეგაც მიიღება ჯგუფის სიმბოლოების კოდების ერთობლიობისაგან შემდგარი რიცხვები, რაშიც ცალკეული სიმბოლოსთვის განსაზღვრულია თავისი ინდექსის კოდი:

$$G_{ASCII\{\ell\}}SD_{\{i\cup j\}}F,L=(G_{ASCII\{1\}}SD_{\{i\cup j\}},G_{ASCII\{2\}}SD_{\{i\cup j\}},...,G_{ASCII\{m\}}SD_{\{i\cup j\}})$$

მიღებული კოდების რიცხობრივი მნიშვნელობა გარდაიქმნება სხვა რიცხობრივ მნიშვნელობად და მიიღება ახალი კოდების სიმრავლე. მიღებული სიმბოლოების კოდების გარდაქმნა ხდება შემდეგნაირად. აიღება სიტყვის პირველი სიმბოლოს კოდი $G_{ASCII\{\ell\}}$, გადამოწმდება მისი მდებარეობის არეალი. თუ კოდი მდებარეობს 32-დან 99-ის ჩათვლით, მაშინ შესრულდება შემდეგი მოქმედება: სიმბოლოს კოდს დაემატება პარამეტრის მნიშვნელობა, რომლიც აღვნიშნოთ – p_3. სადაც $96\le p_3\le 156$. ხოლო, თუ კოდი მდებარეობს 100-დან 126-ის ჩათვლით, მაშინ სიმბოლოს კოდს დაემატება პარამეტრის მნიშვნელობა, რომლიც აღვნიშნოთ – P_4. სადაც $28\le p_4\le 129$.

თუ $32\le G_{ASCII\{\ell\}}F,L\le 99$, მაშინ

$$P_{ASCII\{g\}}F,L=G_{ASCII\{\ell\}}F,L+p_3$$

თუ $100\le G_{ASCII\{\ell\}}F,L\le 126$, მაშინ

$$P_{ASCII\{g\}}F,L=G_{ASCII\{\ell\}}F,L+p_4$$

$$g=g+1;\ \ \ell=\ell+1;\ \ g=\overline{1,t}$$

სადაც $P_{ASCII\{g\}}F,L$ არის სპეციალური სიმბოლოების მოდიფიცირებული კოდებისაგან შემდგარი ერთობლიობის მასივი, რომელიც ფიქსირდება ჯგუფების მიხედვით.

მიღებული ახალი მოდიფიცირებული კოდებით დგება სპეციალური სიმბოლოების ჯგუფი:

$$g = 1$$
$$P_{\{g\}} = P_{ASCII\{\ell\}}F,L$$
$$g = g + 1$$
$$\text{ვიდრე } \; g \le t$$

შედეგად მივიღებთ სპეციალური სიმბოლოების ჯგუფების ერთობლიობას:

$$G_{\{\ell\}}P_{\{g\}}F,L = (G_{\{1\}}P_{\{g\}}, G_{\{2\}}P_{\{g\}}, ..., G_{\{m\}}P_{\{g\}})$$

სპეციალური სიმბოლოებისაგან შემდგარი ჯგუფების გაერთიანებით მიიღება სპეციალური სიმბოლოების ერთობლიობისგან შემდგარი დაშიფრული ინფორმაცია:

$$W_{\{\ell\}}F,L = G_{\{\ell\}}P_{\{g\}}$$
$$\ell = \overline{1,m}\,; \;\; g = \overline{1,t}$$

დაშიფრული ინფორმაცია ცენტრალურ სერვერს მიეწოდება ჯგუფების სახით. თუ პირველი ჯგუფის იდენტიფიკაცია წარმატებით დასრულდა, სერვერი ატყობინებს და ბრძანებას გამოსცემს მეორე ჯგუფის გამოშვებაზე და ა.შ. ჯგუფის ბოლომდე. თუ რომელიმე ჯგუფი არ დაემთხვა სერვერი მაშინვე ბლოკავს აღნიშნულ მომხმარებელს. დაშიფრული ინფორმაციის ამოშიფვრა ცენტრალური სერვერის მიერ ხდება ზემოთგანხილული მეთოდის უკუალგორითმის საშუალებით იგივე პარამეტრების გამოყენებით.

ლიტერატურა:

1. შონია ო., ნარეშელაშვილი გ., ქართველიშვილი ი. უმავთულო ქსელების უსაფრთხოება. სტუ, თბილისი 2009

2. Запечников С.В., Милославская Н.Г., Толстой Л.И. Основы построения виртуальных частных сетей, Москва. 2003

3. Макин Дж.С., Маклин Й. Внедрение, управление и поддержка сетевой инфраструктуры Ms Windows Server 2003, Москва 2004.

ENCIPHERING OF SYMBOLS BY THE COMBINED METHOD IN THE VIRTUAL PRIVATE NETWORK (VPN)

Shonia, Otar, Kartvelishvili Ioseb, Shonia Luka, Beridze Zebur
Georgian Technical University

Summary

This paper presents the combined method of symbols enciphering in the virtual private network. Each stage of a method is in details described. The used encryption and decryption key based on specific parameters is unique and at the entrance to the user's system it changes and doesn't repeat.

ШИФРАЦИЯ СИМВОЛОВ КОМБИНИРОВАННЫМ МЕТОДОМ В ВИРТУАЛЬНОЙ ЧАСТНОЙ СЕТИ (VPN)

Шониа О., Картвелишвили И., Шониа Л., Беридзе З.
Грузинский Технический Университет

Резюме

Представлен комбинированный метод шифрации символов виртуальной частной сети (VPN). Детально описан каждый этап метода. Использованный в системе ключ шифрации и дешифрации на основе специальних параметров является уникальным и при входе систему пользователя он изменяется и не повторяется.