



Разработка системы мониторинга браузера для предотвращения фишинговых атак

Студент: ПЕТРОВ Павел Евгеньевич

Группа: ИКБ-83

Научный руководитель:

К.т.н., доцент каф. ЗСС, ИЗРАИЛОВ Константин Евгеньевич

Актуальность темы исследования и постановка научно-технической задачи

Количество обнаруженных фишинговых ресурсов (Ц.Р. CERT-GIB)



Количество срабатываний системы "Антифишинг" от Kaspersky



Цель: Повышение эффективности применения инструментария для обнаружения фишинга и предупреждения пользователя

Объект: Действия пользователя в интернете

Предмет: Предотвращение потери персональных/корпоративных данных

Задачи:

- Анализ фишинговых сайтов
- Исследование существующих методов выявления фишинговых страниц
- Создание собственного метода
- Разработка ПО для выявления потенциальных угроз
- Проведение экспериментов на прототипе

Признаки фишинговых сайтов

Отсутствие SSL сертификата

Некорректное написание URL-адреса, пытающегося походить на легитимный через малозаметное изменение букв и символов

Отсутствие стандартных компонентов популярных страниц, колонтитулов, полей меню и заголовков

Наличие веб-формы для заполнения, которая, в большинстве случаев, просит ввести персональные данные или конфиденциальную информацию

Рейтинг сайта в интернете среди других страниц

Дата, время и адрес регистрации домена

Требования к разрабатываемой системе

Модульный принцип построения

Централизованное управление системой

Совместное применение автоматического и ручного контроля

Использование принципов машинного обучения

Журналирование

Enter your email

Password

Next

★★★★★

www.amazonn.com/

~~amazon~~

Sign-In

Email or mobile phone number

Continue

By continuing, you agree to Amazon's [Conditions of Use](#) and [Privacy Notice](#).

[Need help?](#)

New to Amazon?

Create your Amazon account

~~Conditions of Use~~ ~~Privacy Notice~~

~~© 1996-2017 Amazon.com, Inc. or its affiliates~~

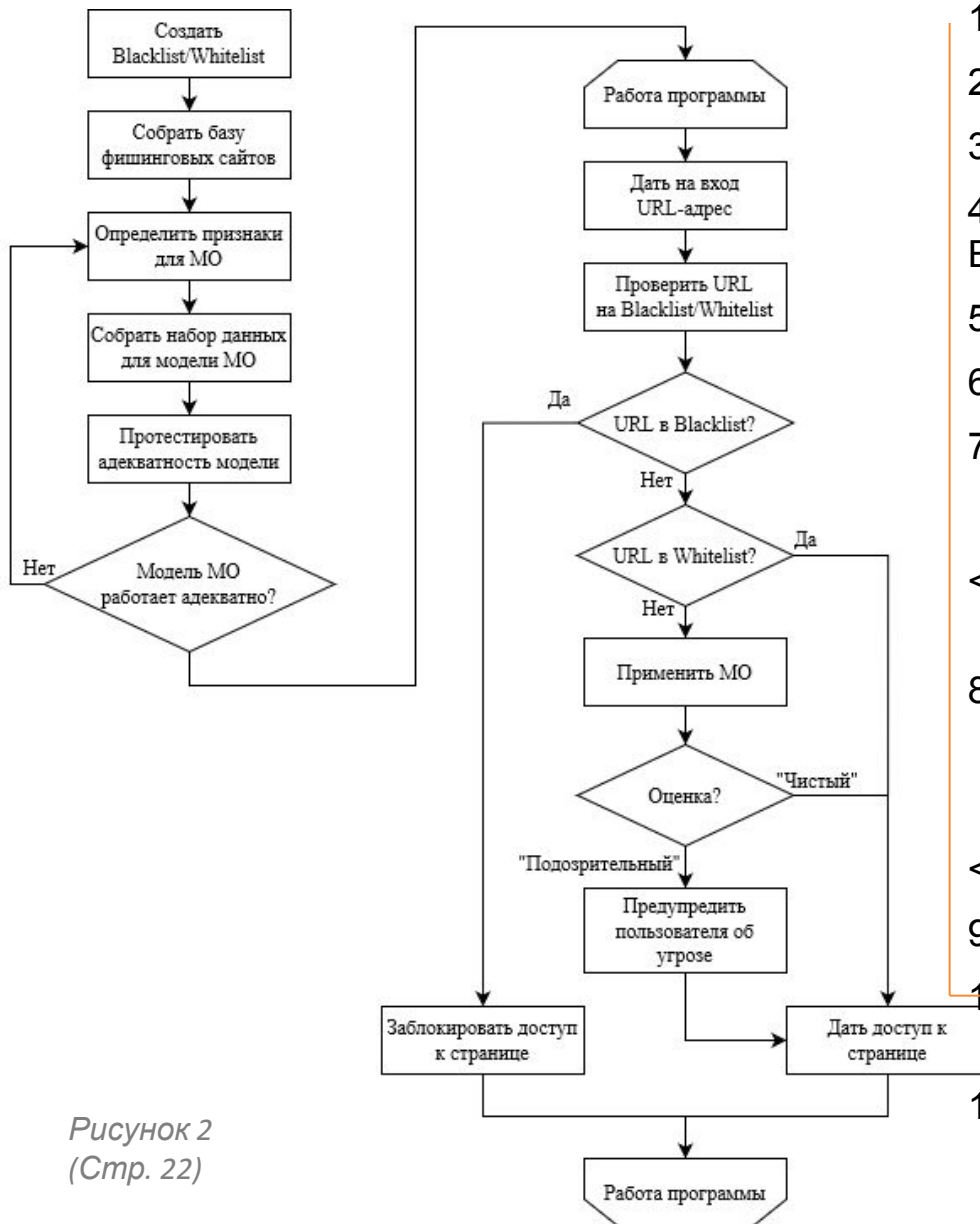
```
▼ domain_age:
  human: "26 years ago"
  timestamp: 845135177
  iso: "1996-10-12T11:46:17-04:00"
```



Название базы данных	Количество URL-адресов	Постоянно обновляется база	Наличие бесплатного API	Бесплатный доступ к базе адресов
OpenPhish	14,7 млн	+	-	-
PhishTank	7,2 млн	+	+	+
PhishStats	4,1 млн	+	+	-
Common Crawl	Ок. 700 тыс.	-	-	+
Phishing.Database	Ок. 630 тыс.	-	-	+

В разрабатываемом прототипе применяются два варианта списков:

- ❑ Корпоративные списки;
- ❑ Списки на основе базы данных адресов.



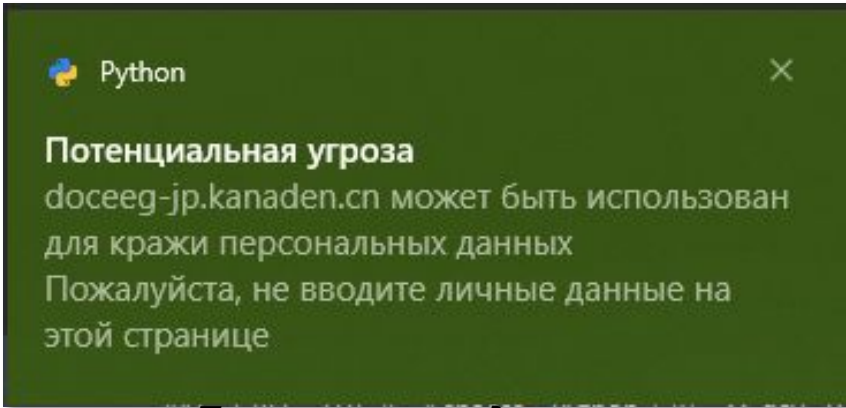
1. $\langle \text{Blacklist, Whitelist} \rangle = \text{Analysis} \langle \text{Company_Politics} \rangle$
2. $\langle \text{DB} \rangle = \text{Comparison} \langle \text{OpenDB_List} \rangle$
3. $\langle \text{Features} \rangle = \text{SelectFeatures} \langle \text{Expert, Data} \rangle$
4. $\langle \text{DataSet} \rangle = \text{Create_DataSet} \langle \text{Features, Expert_Decision} \rangle$
5. $\langle \text{Model} \rangle = \text{Train} \langle \text{Classifier, DataSet} \rangle$
6. $\langle \text{Adequate} \rangle = \text{IsModelGood} \langle \text{Model, TestData} \rangle$
7. If $\langle \text{URL} \rangle$ in $\langle \text{Blacklist} \rangle$: Goto $\langle \text{Шаг 11} \rangle \Rightarrow \langle \text{Block} \rangle$
Else If $\langle \text{URL} \rangle$ in $\langle \text{Whitelist} \rangle$: Goto $\langle \text{Шаг 9} \rangle \Rightarrow \langle \text{Pass} \rangle$
Else: Goto $\langle \text{Шаг 8} \rangle$
8. $\langle \text{Page_Features} \rangle = \text{AnalyzePage} \langle \text{Page} \rangle$
 $\langle \text{Result} \rangle = \text{CheckPage} \langle \text{Model, Page_Features} \rangle$
If $\langle \text{Result} \rangle == \langle \text{Good} \rangle$: Goto $\langle \text{Шаг 9} \rangle \Rightarrow \langle \text{Pass} \rangle$
Else If $\langle \text{Result} \rangle == \langle \text{Bad} \rangle$: Goto $\langle \text{Шаг 10} \rangle \Rightarrow \langle \text{Pass} \rangle + \langle \text{Notify} \rangle$
9. $\langle \text{Pass} \rangle$
10. $\langle \text{Notify} \rangle = \text{ShowNotification} \langle \text{URL, Text} \rangle$
 $\langle \text{Pass} \rangle$
11. $\langle \text{Block} \rangle$

Работа программы завершается при закрытии окна браузера

Рисунок 2
(Стр. 22)

Модульный слой архитектуры

- Модуль инициализации программы;
- Модуль оценки Blacklist/Whitelist;
- Модуль сбора информации о странице;
- Модуль оценки страницы на основе МО;
- Модуль уведомлений;

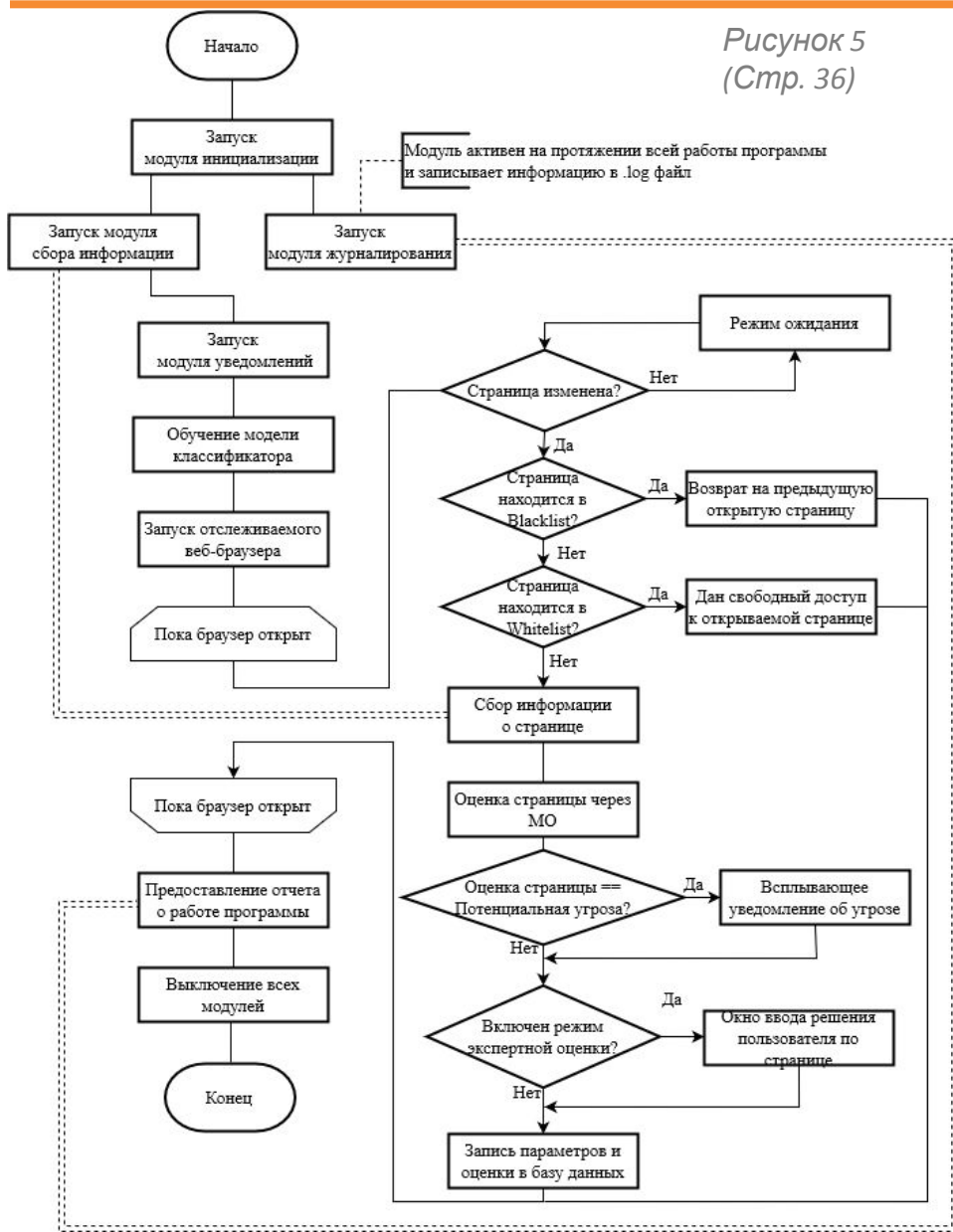


Пример окна уведомления

Информационный слой архитектуры

- Модуль работы с данными

Входные данные	Blacklist/Whitelist	Набор данных для обучения модели классификатора МО	База данных фишинговых сайтов	Список ключевых слов для определения параметров, отправляемых на оценку классификатору
Промежуточные данные	Набор данных, который собирается в процессе работы программы и отправляется заранее обученной модели для итоговой оценки проверяемой страницы			
Выходные данные	Результат оценки страницы моделью	Окно ввода пользовательского решения (экспертный режим работы)		Уведомление о потенциальной угрозе



Общая схема подразделяется на два связанных между собой этапа.

Первый этап относится к работе прототипа до и после непосредственной работы пользователя в окне браузера, можно считать его внешним этапом.

Второй – основной – этап работы программы относится к алгоритмам работы во время использования отслеживаемого браузера. В данном этапе происходит мониторинг открытых страниц, их оценка в Blacklist/Whitelist, работа алгоритмов машинного обучения и уведомление пользователя о потенциальных угрозах.

модулей

для передачи данных

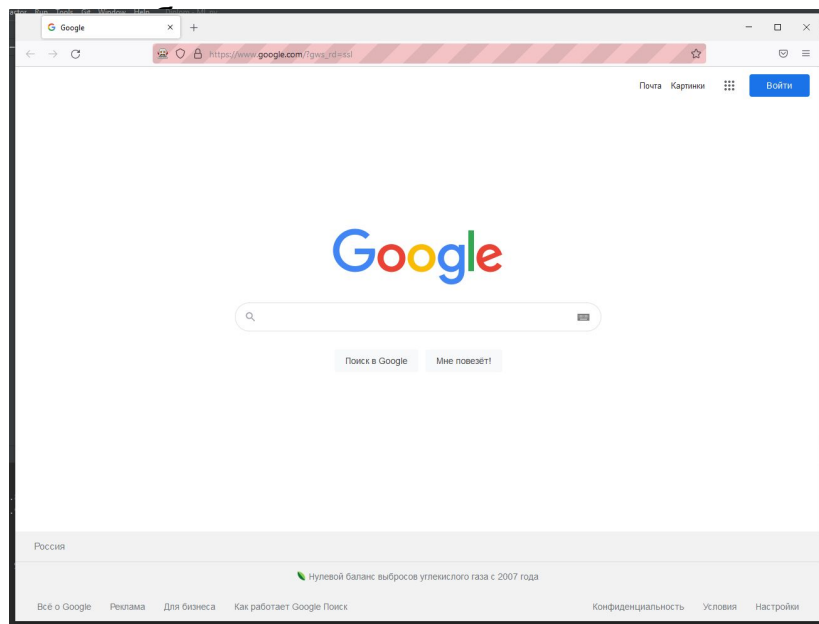
Выбор параметров	Описание
Phishing	<div>«-1» – на странице нет формы для заполнения;</div> <div>«0» – есть форма заполнения, но сайт НЕ в списке фишинговых адресов;</div> <div>«1» – сайт в списке фишинговых адресов.</div>
Rank	<div>«-1» – значение ранга от 0 до 33;</div> <div>«0» – значение ранга от 34 до 66;</div> <div>«1» – значение ранга от 67 до 100.</div>
Suspicious	<div>«-1» – страница в списке подозрительных адресов;</div> <div>«0» – страница не в списке подозрительных адресов;</div> <div>«1» – страница в списке подозрительных адресов.</div>
Password	<div>«0» – страница не в списке подозрительных адресов;</div> <div>«1» – страница в списке подозрительных адресов.</div>
Login	<div>«0» – страница не в списке подозрительных адресов;</div> <div>«1» – страница в списке подозрительных адресов.</div>
Iframe	<div>«0» – страница не в списке подозрительных адресов;</div> <div>«1» – страница в списке подозрительных адресов.</div>
IQS_Phishing	<div>«False» – страница не в списке подозрительных адресов;</div> <div>«True» – страница в списке подозрительных адресов.</div>
IQS_Suspicious	<div>«False» – страница не в списке подозрительных адресов;</div> <div>«True» – страница в списке подозрительных адресов.</div>
IQS_Risk	<div>«-1» – значение ранга от 0 до 33;</div> <div>«0» – значение ранга от 34 до 66;</div> <div>«1» – значение ранга от 67 до 100.</div>
Result	<div>«-1» – сайт чистый;</div> <div>«1» – сайт потенциально опасен.</div>

Базовое тестирование прототипа

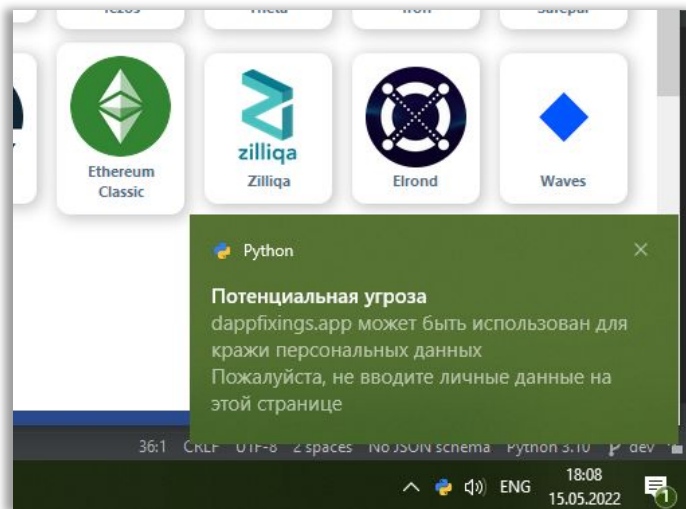
1. Запуск программы и модулей

```
C:\Users\petru\AppData\Local\Programs\Python\Python310\python.exe "E:/PYTHON SMTN/Diplom/scanner.py"  
ML instance is ready  
Notifier online  
Scanner online
```

2. Запуск окна веб



3. Работа модуля уведомлений



4. Работа модуля

Messages	Count	Visited pages:
INFO	15	www.google.com
WARNING	5	www.jetbrains.com
CRITICAL	0	rutube.ru
		www.researchgate.net
		phishtank.org
		dappfixings.app

По окончании работы мы также получаем вывод информации в log-файл

Пошаговая схема

1. Собрать пул адресов (50% - фишинговые страницы; 50% - чистые страницы; в общем случае от 10 до 20 адресов) и листы доступа;
2. Запустить программу;
3. Дождаться запуска отслеживаемого окна веб-браузера;
4. Зайти на URL-адрес, находящийся в Blacklist;
5. Зайти на URL-адрес, находящийся в Whitelist;
6. Открыть ранее собранный пул и поочередно заходить на каждый из адресов;
7. Проверить получаемую программой оценку по открытой странице;
8. Проверить работу уведомлений (при срабатывании);
9. Закрыть веб-браузер, проверить

Исходные данные

эксперимента

Blacklist: «Youtube.com», «Wikipedia.org»;

Whitelist: «Ok.ru», «Vk.com».

Пул URL-адресов для эксперимента:

URL-адрес	Экспертная оценка сайта
Google.com	Чистый
www.sut.ru	Чистый
https://www.twitch.tv/	Чистый
github.com	Чистый
web.telegram.org	Чистый
http://mail.ru/	Чистый
avito.ru	Чистый
http://www.airsoftclub.ru	Чистый
www.aeocsen-aesonm.tkfixuh.ne.pw	Фишинговый
http://metamasf.cc	Фишинговый
https://bitpandac.net/	Фишинговый
sec-tool.net	Фишинговый
http://solanart.ltd	Фишинговый
www.paxfulport.com	Фишинговый
www.pockchain.net/	Фишинговый
www2.aenoeuon.icu	Фишинговый

Ход эксперимента

Шаг 6. Открыть ранее собранный пул и поочередно заходить на каждый из адресов

URL-адрес	Экспертная оценка сайта	Программная оценка сайта
Google.com	Чистый	Чистый
www.sut.ru	Чистый	Чистый
https://www.twitch.tv/	Чистый	Чистый
www.cyberforum.ru	Чистый	Фишинговый
web.telegram.org	Чистый	Чистый
http://mail.ru/	Чистый	Чистый
avito.ru	Чистый	Чистый
http://www.airsoftclub.ru	Чистый	Фишинговый
www.aeocsen-aesonm.tkfixuh.ne.pw	Фишинговый	Фишинговый
http://metamasf.cc	Фишинговый	Фишинговый
https://bitpandac.net/	Фишинговый	Чистый
sec-tool.net	Фишинговый	Фишинговый
http://solanart.ltd	Фишинговый	Фишинговый
www.paxfulport.com	Фишинговый	Фишинговый
www.pockchain.net/	Фишинговый	Фишинговый
www2.aenoeuon.icu	Фишинговый	Фишинговый

Матрица ошибок
Таблица оценки по открытию

$$TP = 7 \quad FP = 1$$

$$FN = 2 \quad TN = 6$$

		Экспертная оценка	
		Фишинговый	Чистый
Оценка модели	Фишинговый	7	1
	Чистый	2	6

Точность:

$$PR = \frac{TP}{TP + FP} = \frac{7}{7 + 1} = 0.875 \Rightarrow 87.5 \%$$

Полнота:

$$RC = \frac{TP}{TP + FN} = \frac{7}{7 + 2} = 0.7(7) \Rightarrow 77.8 \%$$

F-мера:

$$F = 2 \frac{PR * RC}{PR + RC} = 2 \frac{0.875 * 0.777}{0.875 + 0.777} = 0.823 \Rightarrow 82.3 \%$$

Итоговая оценка качества:

$$AC = \frac{P}{N} = \frac{13}{16} = 0.8125 \Rightarrow 81.25 \%$$

Сравнительный анализ прототипа с существующими решениями

Критерии эффективности (формульное представление):

1. $Эфф = \langle Рез_Опер, Рез_Эк \rangle$ где «Эфф» – Эффективность; «Рез» – Результативность; «Опер» – Оперативность; «РесЭк» – Ресурсоэкономность.
2. Применение модели машинного обучения для принятия решения по проверяемой странице;
3. $Рез = \langle Рез_Кач, Рез_Колич \rangle$ где «Рез Кач» - Качественная результативность (Критерии); «Рез_Колич» - Количественная результативность (% Точности).
4. Возможность использования в корпоративных сетях;
Берем «Рез_Колич» в усредненном значении, далее рассчитаем «Рез_Кач»:

$Рез_Кач = \langle Кр_1, \dots Кр_8 \rangle$, где «Кр_1», ... «Кр_8» - Ранее определенные критерии.

$$Эфф_Прототип > Эфф_Аналог(1..4)$$

5. Возможность применения дополнительной экспертной оценки со стороны пользователя;

Решение	Кр_1	Кр_2	Кр_3	Кр_4	Кр_5	Кр_6	Кр_7	Кр_8	Точность обнаружения угрозы
Фильтр Firefox	+	—	—	—	—	—	—	+	~ 73%
Фильтр IE	+	—	—	—	—	—	—	+	~ 70 %
FireEye EX	+	+	+	+	—	—	—	—	~ 87 %
Анти-фишинг в KIS	+	—	+	—	—	—	—	+	~ 91%
Прототип	+	+	+	+	+	+	+	+	~ 81.3 %

СПб ГУТ)))



Благодарю за внимание!

Вопросы?