Судя по схеме у нас MVC (Model View Controller). Значит пользователь приходит на view (html страницу), отправляет данные на контроллер, а контроллер в свою очередь обращается к серверам (Менеджерам), которые в свою очередь идут в БД.

У нас соединение по TCP 389. Так полагаю, что незащищенное. Надо, чтобы контроллер работал на 443 порту. Так же необходимо сделать сделать проверку на браузер клинета, не является ли он устаревшим. Это нужно для исключения таких атак как arp-spoofing, dhcp-spoofing, dns-spoofing, ssl-strip, ssl-split. Необходимо проверять сертификат при протоколе TLS. Сделать рассылку отчетов и уведомлений с доверенным сертификатом, чтобы не попадать в спам. Сертификат доверенного сайта отличит нас от клонов и злоумышленников. При возникновении DOS и DDOS атак, сделать проверку на робота (установить капчу на сайте при перегрузке домена). В контроллере обязательно фильтровать данные от клиентов на спец символы, а так же по длине. Если контроллер написан на интерпретируемом языке может быть инъекция (XSS, htmli, php, sqli, ldapi). При написании контроллера на компилируемом языке может возникнуть переполнение буфера.

Поэтому следить за используемой памяти, чтобы не возникли бинарные уязвимости. Обезопасить запрос к контроллеру от ssrf и csrf. SSRF можно запретить, если сделать обращение к внутренним API невозможным из-вне либо при помощи аутентификации. CSRF устранить при помощи криптостойкого anti csrf token. Приложение сервера запустить от имени пользователя!!! Ни в коем случае не от root на linux или админа на windows, так как это может привести к полной компрометации сервиса. Трафик приложений не должен утекать вне внутренней сети. Для этого надо использовать роутер с поддержкой wireless, а так же NAT. При обращении серверов к БД использовать библиотеки прослойки исключающие инъекции. При извлечении данных вторично проверять данные на спец символы. Сделать недоступными все сервера и контроллеры из-вне по ssh или telnet.

Для сети 1. Сделать подключение по ssh по ключу или хотя бы по 12 символьному паролю. Трафик приложения между работником и кластером реверс прокси сделать по 443 порту. С сетью 2 сделать все аналогично. Настроить waf на защиту от вторжения в периметр.

Для сегмента Internet. Пользоваться только доверенным vpn. Трафик приложений между vpn пользователем по 443 порту. На интернет шлюзе настроить NAT и wireless. Сделать доступными из вне только ПК со статическим ip.

Кластер реверс прокси поддержовать соединение только по 443 порту. Заменить vpn концентратор (hub) на коммутатор (switch) или сетевой мост. На маршрутизаторе настроить NAT и wireless.

Сеть удаленного офиса настроить firewall при подключении из вне. Разрешить подлючаться к офисным ПК только со статическим ip. Настроить NAT и Wireless. Обновить настройки маршрутизатора Cisco и отключить 8 значный код вместо пароля. Наличие сертификата cisco у роутера. Трафик внутри сети сделать защищенным. И настроить сетевую политику роутера и приложений исключающую arp-spoofing. Реверс прокси сделать по https плюс добавить аутентификацию. Сделать белый список ip адресов или хотя бы черный. Настроить права для разных пользователей.

1\*

Необходимо внутри периметра и в некоторых других сегментах установить антивирусное ПО, а так же систему обнаружения вторжений и расследования инцидентов.

В качестве антивирусного ПО подойдут Kaspersky Total Security, Kaspersky Endpoint Security, DR Web, Avast. Необходимо включить проверку файлов и сети. Настроит сигнатурный и эвристический анализ. Можно поставить шифрование на диск AES – такой модуль есть при Kaspersky Endpoint Security. Если не удается установить данные антивирусники, то можно использовать Microsoft Defender, но при этом настроить в нем firewall.

Waf нужен для:

Изначально термин firewall (брандмауэр, экран) обозначал сетевой фильтр, который ставится между доверенной внутренней сетью и внешним Интернетом (отсюда прилагательное «межсетевой»). Этот фильтр был призван блокировать подозрительные сетевые пакеты на основе критериев низких уровней модели OSI: на сетевом и канальном уровнях. Иными словами, фильтр учитывал только IP адреса источника и назначения, флаг фрагментации, номера портов.

Следующим поколением защитных экранов стали системы обнаружения и предотвращения вторжений (IDS/IPS). Они способны изучать в TCP-пакетах поле данных и осуществлять инспекцию на уровне приложения по определённым сигнатурам. Системы IDS приспособлены к тому, чтобы выявлять атаки не только снаружи, но и внутри сети за счет прослушивания SPAN-порта коммутатора.

Если говорить совсем просто, то веб-приложения отличаются от обычных приложений двумя вещами: огромным разнообразием и значительной интерактивностью. Это создаёт целый ряд новых угроз, с которыми традиционные межсетевые экраны не справляются: по нашим оценкам, в 2014 году 60% атак на корпоративные сети осуществлялись через веб-приложения, невзирая на наличие традиционных защитных средств.  
  
Именно здесь вступает в дело Web Application Firewall (WAF), защитный экран для приложений, осуществляющих передачу данных через HTTP и HTTPS. Вот какие функции отличают WAF от защитных систем предыдущих поколений: